



Pembangunan Sistem Informasi Penjadwalan *Reminder* Tagihan Airtime Pelanggan VMS Berbasis Web

Lestari Margatama¹⁾, Indra Riyanto²⁾

¹⁾Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur
Jl. Cileduk Raya Petukangan Utara, Jakarta Selatan, 12260

²⁾Program Studi Teknik Elektro, Fakultas Teknik, Universitas Budi Luhur
Jl. Cileduk Raya Petukangan Utara, Jakarta Selatan, 12260

E-mail : lestari.margatama@budiluhur.ac.id¹⁾; indra.riyanto@budiluhur.ac.id²⁾

Abstract

Recording of the current billing period is still done by the Finance department manually using spreadsheet software, so that with the number of customers more and more work becomes ineffective and often delays billing to customers after the obligation to install tracking devices on ships by the Ministry of Maritime Affairs and Fisheries. The average bill every month is 650 bills to handle 900 customers with 2/3 composition using a monthly subscription scheme while the billing capacity is 640 bills per month, so there is a delay of about 10 bills. Based on this, the purpose of this study is build a system of scheduling and reminder airtime billing to web-based VMS customers, as a form of reminder model in order to help improve the effectiveness of activities. This application is made with the CodeIgniter framework, bootstrapping and using a MySQL database. Modeling the system using UML (Unified Modeling Language). The method used is the Rapid Application Development (RAD) method. The results of this research are the establishment of a web-based VMS customer airtime bill scheduling and reminder system that can increase the effectiveness of billing activities from the beginning there are an average of 10 late billing, become fully collectible and the trial results have 100% valid results, all functions are running good and as expected.

Keywords: *Reminder System, Rapid Application Development, Billing System, Vessel Monitoring And Tracking System.*

Abstrak

Pencatatan periode tagihan saat ini masih di lakukan oleh bagian *Finance* secara manual menggunakan *software spreadsheet*, sehingga dengan jumlah pelanggan yang semakin banyak pekerjaan menjadi tidak efektif dan seringkali terjadi keterlambatan penagihan kepada pelanggan setelah diberlakukannya kewajiban memasang alat pelacak pada kapal oleh Kementrian Kelautan dan Perikanan. Rata-rata tagihan setiap bulannya 650 tagihan untuk menangani 900 pelanggan dengan komposisi 2/3-nya menggunakan skema berlangganan bulanan sedangkan kapasitas penagihan adalah 640 tagihan per bulannya, maka terjadi keterlambatan tagihan sekitar 10 tagihan.. Berdasarkan hal tersebut, tujuan dari penelitian ini adalah membangun suatu sistem penjadwalan dan reminder tagihan airtime kepada pelanggan VMS berbasis web, sebagai bentuk model pengingat agar dapat membantu meningkatkan keefektifitasan kegiatan. Aplikasi ini dibuat dengan framework CodeIgniter, bootstrap dan menggunakan basis data MySQL. Pemodelan sistemnya menggunakan UML (*Unified Modeling Language*). Metode yang di gunakan adalah metode *Rapid Application Development (RAD)*. Hasil dari penelitian ini adalah terbangunnya sebuah sistem penjadwalan dan reminder tagihan airtime pelanggan VMS berbasis web yang dapat meningkatkan efektifitas kegiatan penagihan dari awalnya terdapat rata-rata 10 keterlambatan penagihan, menjadi dapat tertagih seluruhnya serta hasil uji coba memiliki hasil 100% valid yaitu semua fungsi berjalan baik dan sesuai harapan.

Kata kunci: *Reminder System, Rapid Application Development, Billing System, Vessel Monitoring And Tracking System.*

1. Pendahuluan

Dengan internet aktifitas bisnis apapun akan memudahkan pelaku bisnis dalam mendapatkan

informasi, bahkan sekarang segala macam bentuk informasi sangat cepat dan mudah sekali diperoleh, untuk mendapatkan informasi yang tepat dan relevan, informasi harus diseleksi terlebih dahulu. Sebuah

perusahaan akan mampu bersaing dengan perusahaan lain apabila memanfaatkan sistem informasi ini. Salah satu perusahaan atau jenis usaha yang menerapkan sistem ini yaitu dalam bidang kelautan atau lebih fokus ke bidang penjualan alat GPS dan pengembangan sistem *monitoring* posisi kapal-kapal laut berbasis *web cloud*.

SRS adalah perusahaan yang bergerak dibidang IT, salah satu produknya adalah menyediakan layanan tracking kapal laut yang mencakup penjualan alat pelacakan kapal laut dan penyediaan system monitoring kapal atau *Vessel Monitoring system* (VMS). Saat ini pelanggan yang menggunakan layanan ini bertambah banyak jumlahnya dengan diberlakukannya kewajiban memasang alat pelacak pada kapal oleh Kementerian Kelautan dan Perikanan. Posisi kapal-kapal laut tersebut dipantau melalui satelit dan dikirim data nya ke *server* untuk disajikan dalam bentuk aplikasi VMS ke para pelanggan. Setiap menjelang akhir masa periode pemakaian layanan VMS ini, Bagian *Finance* SRS mengirimkan email notifikasi dan tagihan biaya ke pelanggan untuk memperpanjang masa periode pemakaian layanan VMS. Proses selanjutnya adalah mencatat secara manual periode penagihan yang baru. Pencatatan periode tagihan saat ini masih di lakukan secara manual menggunakan *software spreadsheet* sehingga dengan jumlah pelanggan yang semakin banyak pekerjaan menjadi tidak efektif dan seringkali terjadi keterlambatan penagihan kepada pelanggan setelah diberlakukannya kewajiban memasang alat pelacak pada kapal oleh Kementerian Kelautan dan Perikanan. Rata-rata tagihan setiap bulannya 650 tagihan untuk menangani 900 pelanggan dengan komposisi 2/3-nya menggunakan skema berlangganan bulanan sedangkan kapasitas penagihan adalah 640 tagihan per bulannya, maka terjadi keterlambatan tagihan sekitar 10 tagihan. Keterlambatan ini dapat merugikan perusahaan karena selama rentang waktu tersebut perusahaan tetap harus membayar biaya *satellite airtime* ke *provider* meskipun pelanggan belum membayar masa perpanjangan.

Perkembangan teknologi memberikan manfaat yang besar bagi kegiatan manusia sehingga menuntut manusia agar dapat memahami teknologi dan juga memanfaatkan teknologi tersebut untuk menunjang kegiatan proses bisnis yang ada. Penggunaan sistem pengingat otomatis untuk mengingatkan tanggal kadaluwarsa sudah banyak dilakukan sebagai pengingat untuk berbagai kegiatan, bahkan dapat digabungkan dengan sistem *calendar* pada perangkat. Secara umum, sistem pengingat dapat dibagi menjadi antara lain menjadi: *task reminder*, *location reminder*, *medical reminder*, dan *activity reminder*. Pembuatan sistem pengingat untuk acara menggunakan OCR sehingga dapat melakukan tugasnya secara otomatis dengan membaca dokumen, melakukan ekstraksi informasi, dan memasukkan

informasi jadwal acara tersebut ke aplikasi kalender. Kelebihan sistem ini adalah dapat mengenali teks tanpa membutuhkan sistem pengenalan tertentu seperti *barcode* atau *QR-code* sehingga dapat mengenali penjadwalan yang berupa teks ataupun tulisan [1].

Penelitian ini bertujuan membangun sistem informasi penjadwalan pengingat tagihan kepada pelanggan berbasis web menggunakan *framework CodeIgniter* dan database *MYSQL*.

Rapid Application Development atau RAD adalah salah satu jenis pemodelan proses yang diawali dengan pembuatan prototype secara cepat (*rapid prototyping*) untuk memperoleh ciri-ciri dasar dari sebuah sistem yang dibangun. RAD dirintis oleh James Martin pada awal 90-an untuk membuat sebuah model yang fleksibel dan dapat disesuaikan dengan kebutuhan yang berubah-ubah. Keterbatasan yang dimiliki oleh RAD adalah berkurangnya *scalability* apabila akan mengubah ukuran sistem dan kualitas sistem yang menurun. Penggantian fase manual dalam pengumpulan kebutuhan menjadi proses otomatis dengan menggunakan teknik agen kecerdasan tiruan (*artificial intelligence agent*) [2].

Penerapan RAD di Indonesia dalam tiga tahun terakhir kebanyakan adalah untuk pengembangan website sistem inventaris atau perancangan *website inventory* [3], sistem *service guide* atau *website service guide* [4], dan untuk sistem akademik, sistem yang dibuat adalah dengan proses iterasi pembuatan sistem informasi perpustakaan yang mencakup transaksi buku, data anggota, serta perhitungan denda keterlambatan [5]. RAD juga digunakan untuk penjadwalan kegiatan belajar mengajar di sekolah [6] dan Sistem Informasi Akademik untuk kegiatan dosen dan mahasiswa [7].

Reminder atau Sistem Pengingat adalah sebuah sistem pesan yang membantu seseorang untuk mengingatkan sesuatu hal. Sistem Pengingat dapat lebih bermanfaat ketika informasi kontekstual digunakan untuk menyajikan informasi pada waktu yang tepat dan tempat yang tepat. Sistem Pengingat dapat digunakan dalam manajemen waktu yang berfungsi untuk memberi alarm peringatan berupa pemberitahuan berbasis lokasi, waktu maupun catatan yang bersifat kontekstual [8].

Penelitian sebelumnya untuk mengurangi beban kerja manual, beberapa aplikasi menggunakan teknologi sebagai pengingat dan pendeteksi objek yang mungkin hilang atau tertinggal. Sistem ini menggunakan taksonomi dua tingkat, yaitu *class* dan *object*. Sistem pengingat sebagai *task reminder* umumnya terhubung dengan aplikasi kalender yang terdapat pada kedua belah pihak (pengirim maupun penerima) sehingga dapat otomatis membuat atau memperbarui jadwal [9].

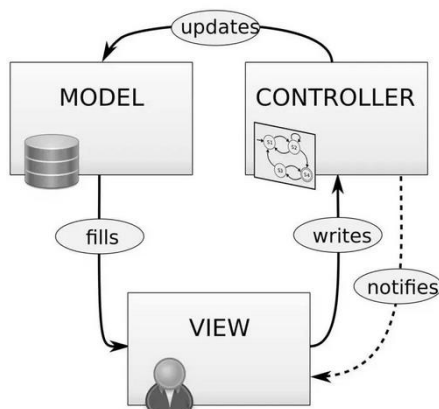
kebanyakan sistem pengingat saat ini digunakan pada aplikasi *mobile* untuk keperluan medis, antara lain aplikasi kompleks menggunakan aplikasi khusus, MMS atau SMS untuk penjadwalan perjanjian dengan dokter, pemberitahuan informasi medis, atau pemberitahuan sederhana untuk penjadwalan imunisasi [10-12].

Penelitian sebelumnya mengenai hasil penggunaan SMS-reminder dalam layanan *healthcare system* dimana dari 162 kasus yang memenuhi syarat penggunaan sistem pengingat ini, 93 kasus merupakan pengingat medis terkait aktivitas dan pengobatan, dan 56 kasus terkait pengaturan jadwal perjanjian medik [10].

Penelitian sebelumnya mengenai evaluasi efektivitas sistem pengingat ini dalam peningkatan ketaatan pada perjanjian medik, dimana didapatkan temuan bahwa sistem pengingat ini lebih efektif untuk pasien berusia di atas 60 tahun [11].

Penelitian sebelumnya mengenai Pengujian penerapan sistem pengingat. Pada program vaksinasi anak di Ethiopia dapat meningkatkan keikutsertaan keluarga dalam program tersebut [12].

Konsep arsitektur *Model-View-Controller* atau MVC adalah istilah yang paling sering disebutkan dalam dunia *web-programming* beberapa tahun terakhir ini. Orang-orang yang bersinggungan secara langsung atau pun tidak langsung dengan dunia pembangunan aplikasi berbasis web pasti pernah mendengar istilah MVC, dikarenakan kepopuleran dan urgensitasnya. Gambar 1 menunjukkan interaksi dari komponen-komponen pada MVC [13].



Gambar 1. Diagram interaksi antar komponen MVC [13]

MVC adalah konsep arsitektur dalam pembangunan aplikasi berbasis web yang membagi aplikasi web menjadi 3 bagian besar. Yang mana setiap bagian memiliki tugas-tugas serta tanggung jawab masing-masing. Tiga bagian tersebut adalah: *model*, *view* dan *controller*. **Model**: Bertugas untuk mengatur, menyiapkan, memanipulasi dan mengorganisasikan data dari *database* sesuai dengan instruksi dari *controller*. **View**: Bertugas untuk

menyajikan informasi yang mudah dimengerti kepada *user* sesuai dengan instruksi dari *controller*. **Controller**: Bertugas untuk mengatur apa yang harus dilakukan oleh *model* serta *view* mana yang harus ditampilkan berdasarkan permintaan dari *user*. Namun, terkadang permintaan dari *user* tidak selalu memerlukan aksi dari *model* [14].

Unified Modeling Language atau lebih sering dikenal dengan istilah UML merupakan alat bantu yang digunakan dalam merancang serta membangun bahasa pemodelan yang berorientasi objek. Secara umum uml lebih mudah dimengerti karena memberikan pemodelan yang eksplisit dan mudah dikembangkan dan dapat bertukar model secara spesifik dan visualisasi. **Use Case Diagram**: merupakan sebuah teknik atau rangkaian yang berkelompok dan saling berkaitan sehingga membentuk struktur rangkaian yang teratur. **Sequence Diagram**: menggambarkan tentang objek-objek *use case* dengan cara mendeskripsikan waktu hidup objek dan *message* yang dikirim serta diterima [15].

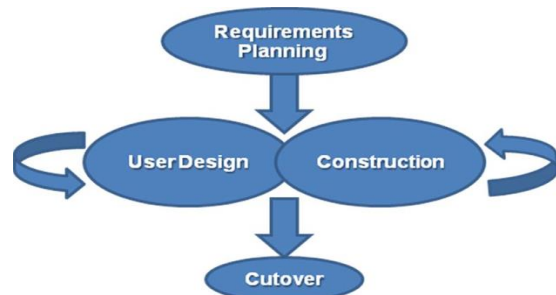
Class Diagram: menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun *sistem*. **Activity Diagram**: sebuah *sophisticated data flow diagram* (DFD) yang digunakan pada analisis struktural dapat dilihat pada *activity diagram*. Berbeda dengan DFD, suatu proses bisnis dapat dimodelkan menjadi perilaku yang digunakan dalam *activity diagram*. *Activity diagram* difungsikan sebagai proses pengambil keputusan yang kompleks dan bersamaan dengan aktifitas yang berlangsung sehingga mempunyai notasi untuk pemodelan [16].

Aliran sistem informasi: merupakan arus yang *mengarikan* seluruh bagan dari sebuah sistem [17].

RAD merupakan metode yang memfokuskan pada kecepatan dalam pengembangan sistem untuk memenuhi kebutuhan pengguna atau pemilik sistem seperti prototyping namun mempunyai cakupan yang lebih luas [18].

2. Metode Penelitian

Metode Penelitian yang digunakan untuk membuat aplikasi penjadwalan ini yaitu dengan menggunakan *Rapid Application Development* (RAD), seperti Gambar 2.



Gambar 2. Empat fase dari model RAD

Alur kerja RAD hanya dibagi menjadi empat tahap yaitu:

2.1 Requirements Planning

Tahap awal melakukan pemenuhan kebutuhan pengguna. Dalam fase ini semua kebutuhan pengguna dikumpulkan dan dicatat. Pada fase ini, pengguna dan penganalisis bertemu untuk mengidentifikasi tujuan-tujuan aplikasi atau sistem serta untuk mengidentifikasi syarat-syarat informasi yang ditimbulkan dari tujuan-tujuan tersebut. Orientasi dalam fase ini adalah menyelesaikan masalah-masalah perusahaan. Pengembang sistem harus dapat mencari informasi untuk memenuhi kebutuhan aplikasi ini. Berdasarkan hasil wawancara yang dilakukan peneliti dengan bagian marketing dan bagian finance bahwa cara yang digunakan dalam kegiatan penagihan pelanggan masih dilakukan dengan cara mencatat secara manual. Alur kerja dari sistem manual yang sedang berjalan adalah sebagai berikut:

1. Bagian Marketing memberikan data pelanggan baru ke Bagian Finance.
2. Bagian Finance melakukan penagihan ke pelanggan dan mencatat data pelanggan baru ke database.
3. Bagian Finance melakukan proses penagihan ke pelanggan dengan membuat tagihan dan mengirimkan email reminder ke pelanggan secara manual.
4. Bagi pelanggan yang sudah melakukan pembayaran, bagian Finance memperbarui tanggal jatuh tempo penagihan pada file spreadsheet.
5. Bulan dan tahun berikutnya Bagian Finance melakukan penagihan berdasarkan sorting manual secara periodik.

Kebutuhan pengguna yaitu dalam hal ini bagian marketing dan bagian finance terhadap sistem yang akan dibangun ada pada Tabel 1.

Tabel 1. Kebutuhan Pengguna

No	Pengguna	Kebutuhan
1	Bagian Marketing	Adanya data pelanggan dan aset secara terpusat. Dapat melihat status data aset secara otomatis Dapat melihat history aset secara otomatis Dapat membuat laporan jumlah dan status pelanggan secara otomatis.
2	Bagian Finance	Adanya data pelanggan dan aset secara terpusat Sistem diharapkan dapat menampilkan data tagihan setiap bulannya dalam tahun berjalan baik itu tagihan bulanan maupun tagihan tahunan. Sistem dapat membuat tagihan dan mengirimkan email reminder ke pelanggan

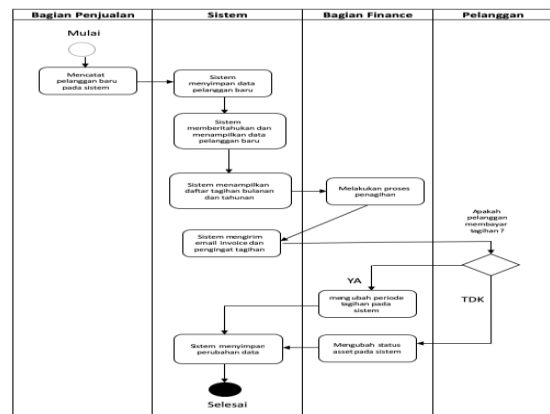
Sistem dapat menyimpan perubahan data dan history aset.
 Sistem dapat membuat laporan jumlah tagihan pelanggan.

2.2 User Design

Pada tahap kedua masih melibatkan semua pihak, yaitu proses mendesain sistem perangkat lunak sesuai dengan kebutuhan pengguna. Pembuatan sistem menggunakan pemodelan sistem berorientasi objek melalui diagram activity, use case, dan class.

2.1.1 Activity Diagram

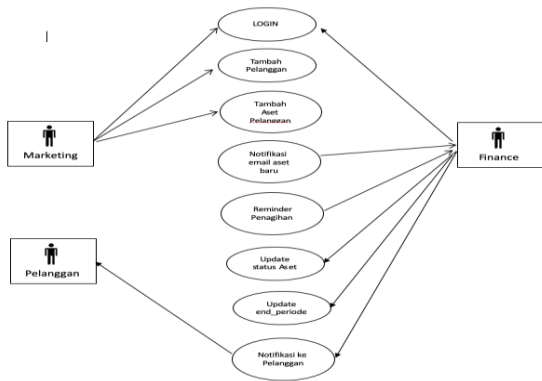
Activity Diagram merupakan rancangan aliran aktivitas atau aliran kerja dalam sebuah sistem yang akan dijalankan. Activity Diagram juga digunakan untuk mendefinisikan atau mengelompokkan aliran tampilan dari sistem tersebut. Activity Diagram memiliki komponen dengan bentuk tertentu yang dihubungkan dengan tanda panah. Panah tersebut mengarah ke-urutan aktivitas yang terjadi dari awal hingga akhir. Pada gambar 3 merupakan activity diagram yang akan di usulkan pada pembangunan sistem informasi penjadwalan pengingat tagihan *airtime* pelanggan VMS.



Gambar 3. Activity Diagram

2.1.2 Use Case Diagram

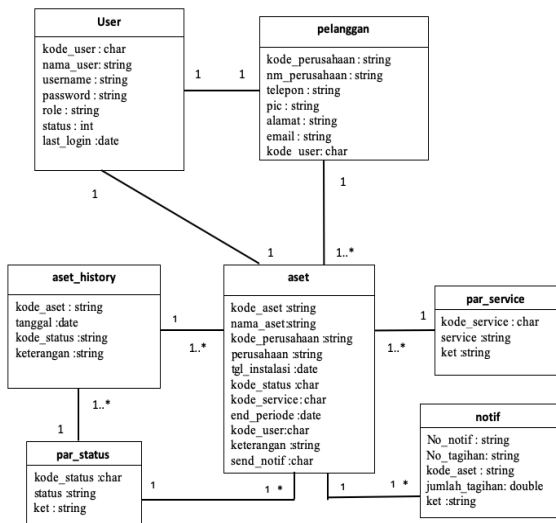
Use Case Diagram memperlihatkan hubungan yang terjadi antara aktor-aktor, dalam hal ini bagian marketing, bagian finance, dan pelanggan, dengan use case dalam sistem yang akan dibuat. Salah satu manfaat dari diagram use case adalah untuk memperjelas alur komunikasi antar bagian sistem. Calon pengguna sistem dapat mengamati diagram use case untuk mendapatkan pengalaman yang utuh tentang sistem yang akan dikembangkan.



Gambar 4 Use Case Diagram

2.1.3 Class Diagram

Class Diagram atau Diagram Kelas menggambarkan struktur kelas di dalam sistem. Class merepresentasikan sesuatu yang ditangani oleh sistem. Diagram kelas dapat membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem. Gambar 5 menunjukkan hasil klasifikasi dari sistem informasi penjadwalan tagihan pelanggan yang dibangun.

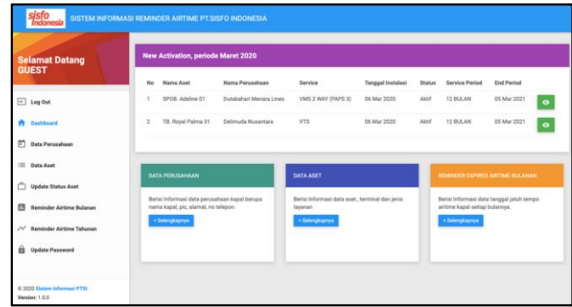


Gambar 5. Class Diagram sistem penjadwalan tagihan

2.3 Construction

Construction yaitu tahap pembangunan dan implementasi sistem. Pada tahap ini sistem mulai dibangun sesuai dengan kebutuhan pengguna dan aspek-aspek pada workshop desain yang telah disepakati. Pada sistem ini dibuat dengan bahasa pemrograman CodeIgniter dan menggunakan basis data MySQL.

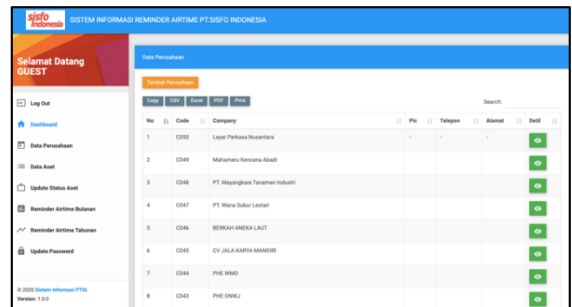
Sistem yang telah dirancang kemudian dibuat dan diimplementasikan ke pengguna, hasil penelitian menampilkan beberapa bagian dari sistem, yaitu: laman dashboard, laman pelanggan, laman aset pelanggan, laman penjadwalan tagihan, laman notifikasi, dan laman update status.



Gambar 6. Laman Dashboard

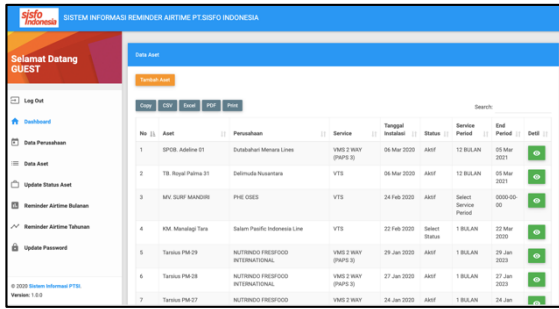
Gambar 6 merupakan implementasi laman dashboard yang menampilkan seluruh daftar menu yang dapat digunakan oleh user dan informasi dasar yang dapat digunakan oleh user. Panel sebelah kiri adalah daftar menu yang dapat digunakan oleh user, panel tengah berisi informasi aset terbaru, panel bawah berisi fungsi utama dari sistem ini.

Gambar 7 merupakan halaman menu pelanggan yang menampilkan seluruh daftar perusahaan yang menjadi pelanggan. Pelanggan adalah para pemilik kapal yang menggunakan layanan VMS. Data perusahaan atau pelanggan yang baru hanya dapat di input oleh bagian admin marketing. Pada laman ini terdapat fasilitas pencarian pelanggan dan fasilitas import data dalam bentuk file Excel, CSV, dan PDF.



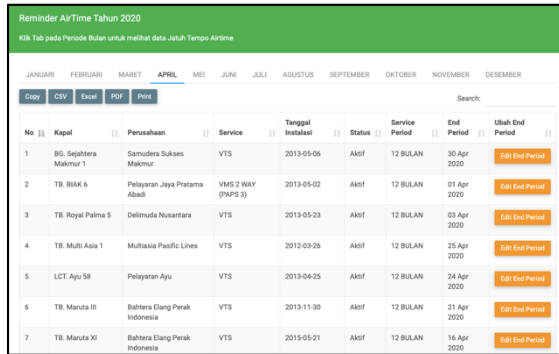
Gambar 7. Laman Menu Pelanggan

Gambar 8 merupakan laman menu Aset yang menampilkan seluruh daftar aset pelanggan. Satu perusahaan atau pelanggan dapat mempunyai lebih dari satu aset. Data aset berupa nama kapal, nama perusahaan, nama service atau paket yang di gunakan, tanggal instalasi alat pelacakan kapal, dan end period yaitu tanggal habis berlangganan layanan VMS. Data aset ini hanya dapat diinput oleh admin sistem di bagian marketing. Setelah melakukan input data aset yang baru maka sistem akan mengirimkan email notifikasi ke bagian finance yang memberitahukan bahwa ada pelanggan baru yang harus ditagih. Pada halaman ini terdapat fasilitas pencarian data aset dan fasilitas import data dalam bentuk Excel, CSV, dan PDF.

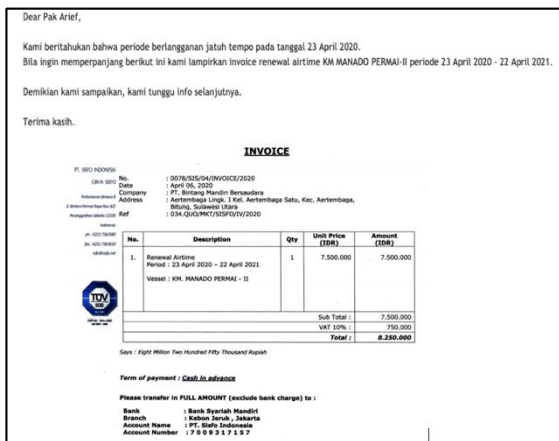


Gambar 8. Halaman Data Aset Pelanggan

Laman penjadwalan tagihan pelanggan pada Gambar 9 menampilkan seluruh penjadwalan tagihan pelanggan setiap bulannya dalam satu tahun berjalan. Pada laman ini bagian Finance dapat mengirimkan email notifikasi akan habisnya masa berlangganan dan terlampir biaya tagihan untuk periode berikutnya kepada pelanggan. Bila terdapat banyak tagihan dalam satu periode maka dapat terjadi keterlambatan penagihan karena waktu pencarian dan pencatatan yang dibutuhkan menjadi lama.

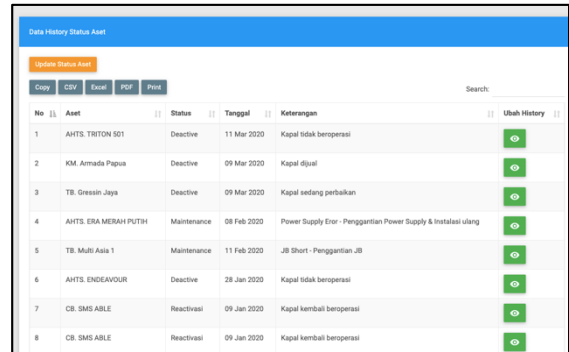


Gambar 9. Halaman Penjadwalan Tagihan Pelanggan



Gambar 10. Email Notifikasi Pelanggan

Gambar 10 merupakan tampilan email notifikasi kepada pelanggan yang dikirim oleh sistem. Bila pelanggan memperpanjang masa berlangganan maka bagian finance mengupdate data tanggal akhir berlangganan. Pada laman ini terdapat fasilitas pencarian dan fasilitas import data dalam bentuk Excel, CSV, dan PDF.



Gambar 11: Halaman Update Status Aset

Laman Update Status Aset yang ditampilkan pada Gambar 11 berisi history data aset pelanggan. Laman ini digunakan untuk memperbarui status aset pelanggan, misalnya bila terjadi kerusakan alat yang memerlukan penggantian alat maka status aset adalah dalam maintenance, sedangkan bila pelanggan ingin berhenti berlangganan maka bagian finance mengupdate status aset pelanggan pada laman ini menjadi non-aktif. Pada halaman ini terdapat fasilitas pencarian dan fasilitas import data dalam bentuk Excel, CSV, dan PDF.

2.4 Cutover

Tahap terakhir pada metode RAD adalah *cutover* yaitu pengujian sistem, *changeover* ke sistem baru dan pelatihan pengguna. Sistem yang telah dibuat telah di implementasikan dan di uji coba oleh pengguna. Metode pengujian yang dipakai adalah metode *black-box* yaitu pengujian yang memperhatikan masukan dan keluaran akhir dari *use case* yang dijalankan.

3. Hasil dan Pembahasan

3.1 Pengujian Sistem

Bagian ini akan memuat hasil pengujian terhadap perangkat lunak yang telah diimplementasikan. dan hasil uji coba memiliki hasil 100% valid yaitu semua fungsi berjalan baik dan sesuai harapan.

Tabel 2. Hasil Pengujian dengan menggunakan black box

Fungsi	Prosedur Pengujian	Hasil
Login	Mengakses halaman login, memasukan username dan password valid untuk SIGN IN	OK
View Dashboard	Sistem menampilkan semua menu yang dapat diakses user pada halaman Dashboard, menampilkan data aset terbaru pada bulan berjalan, dan menampilkan data history Aset	OK
View Perusahaan	Klik menu Data Perusahaan untuk mengakses halaman Data Perusahaan	OK
Create Perusahaan	Klik tombol tambah pada halaman Perusahaan, memasukan data perusahaan kedalam form yang tersedia, menyimpan data perusahaan	OK
Update Perusahaan	Klik tombol edit pada halaman Perusahaan, memperbarui data perusahaan, menyimpan perubahan	OK
Delete Perusahaan	Klik tombol delete pada halaman Perusahaan	OK

Pencarian Perusahaan	Masukkan kata kunci pencarian perusahaan, menampilkan hasil pencarian perusahaan	OK
Import Data Perusahaan	Klik tombol CSV atau Excel atau PDF untuk mendownload hasil import sesuai dengan format yang di pilih	OK
View Aset	Klik menu Data Aset untuk mengakses halaman Data Aset	OK
Create Aset	Klik tombol tambah pada halaman Aset untuk memasukan data aset kedalam form yang tersedia	OK
Update Aset	Memperbarui data aset	OK
Delete Aset	Klik tombol delete pada halaman Aset	OK
Pencarian Aset	Masukkan kata kunci pencarian aset Menampilkan hasil pencarian aset	OK
Import Data Aset	Klik tombol CSV atau Excel atau PDF	OK
View History Aset	Sistem mendownload hasil import sesuai dengan format yang di pilih Klik menu View History Aset untuk mengakses halaman View History Aset	OK
Create History Aset	Klik tombol tambah pada halaman View History Aset, memasukkan data history kedalam form yang tersedia Klik tombol simpan untuk menyimpan data history.	OK
Update History	Klik tombol edit pada halaman History Aset, memperbarui data history aset Klik tombol save untuk menyimpan perubahan	OK
Pencarian History	Masukkan kata kunci pencarian history	OK
Import Data History	Menampilkan hasil pencarian history Klik tombol CSV atau Excel atau PDF	OK
View Reminder Airtime Bulanan	Sistem menampilkan hasil download sesuai dengan format yang di pilih Klik menu View Reminder Airtime Bulanan untuk mengakses halaman View Reminder Bulanan.	OK
Kirim email reminder ke pelanggan	Fungsi menampilkan daftar tagihan bulanan dalam tahun berjalan. Pilih bulan tagihan	OK
Update end periode penagihan	Klik tombol kirim email	OK
Pencarian Tagihan Bulanan	Pilih tagihan yang akan di edit, klik tombol edit end period, klik tombol simpan untuk menyimpan perubahan	OK
Import Tagihan Bulanan	Masukkan kata kunci pencarian tagihan bulanan, menampilkan hasil pencarian tagihan bulanan	OK
View Reminder Airtime Tahunan	Klik tombol CSV atau Excel atau PDF	OK
Kirim email reminder ke pelanggan	Sistem mendownload hasil import sesuai dengan format yang di pilih Klik menu View Reminder Airtime Tahunan untuk mengakses halaman View Reminder Tahunan, Fungsi akan menampilkan daftar tagihan tahunan.	OK
Update End Period Tagihan Tahunan	Pilih data tagihan tahunan Klik tombol kirim email	OK
	Pilih tagihan tahunan yang akan di edit	OK
	Klik tombol edit end period	
	Klik tombol simpan untuk menyimpan perubahan	

Pencarian Tagihan Tahunan	Masukkan kata kunci pencarian tagihan tahunan	OK
	Menampilkan hasil pencarian tagihan tahunan	

Sistem ini di mulai implementasikan sejak Oktober 2019, selama periode tersebut telah di lakukan penagihan otomatis sebanyak 1875 penagihan tanpa ada keterlambatan dengan rincian seperti pada Tabel 3.

Tabel 3. Hasil Implementasi Sistem Penagihan

No	Bulan	Jumlah Tagihan	Jumlah tertagih
1	Okt 2019	610	610
2	Nov 2019	650	650
3	Des 2019	615	615

4 Kesimpulan

Kesimpulan didapatkan sebagai berikut aplikasi penjadwalan pengingat tagihan pelanggan berbasis web ini bisa menjadi alat untuk mempermudah proses pencatatan dan pengingat penagihan pada pelanggan, sehingga menjadi lebih efektif yaitu dengan memanfaatkan komputer sebagai alat pengolah data, dapat meminimalkan terjadinya keterlambatan tagihan kepada pelanggan dan dapat mengurangi waktu yang dibutuhkan untuk membuat tagihan, selama tiga bulan implementasi sistem tidak terdapat keterlambatan penagihan, integrasi data antara divisi dalam perusahaan sehingga dapat mempermudah dalam pemutakhiran data dan Hasil dari penelitian ini adalah terbangunnya sebuah sistem penjadwalan dan reminder tagihan airtime pelanggan VMS berbasis web yang dapat meningkatkan efektifitas kegiatan penagihan dari awalnya terdapat rata-rata 10 keterlambatan penagihan, menjadi dapat tertagih seluruhnya, serta hasil uji coba memiliki hasil 100% valid yaitu semua fungsi berjalan baik dan sesuai harapan

Saran yang perlu diajukan untuk penelitian selanjutnya perlu adanya pengembangan fitur berbasis *smartphone*.

Daftar Pustaka

- [1] Boubashi, F., 2019, From Poster to Mobile Calendar: An Event Reminder using OCR, *International Journal of Advanced Computer Science and Application*, 10(10), pp. 580-585.
- [2] Fatima, F., Javed, M., Amjad, F., dan Khan, U.G., 2018. An Approach To Enhance Quality Of The Rad Model Using Agents. *Journal of American Science*, 14(9), pp. 47-55.
- [3] Irnawati, O. dan Listianto, G.B.A., 2018. Metode Rapid Application Development (RAD) pada Perancangan Website Inventory PT. SARANA ABADI MAKMUR BERSAMA (S.A.M.B) JAKARTA. *Jurnal Evolusi*, 6(2), pp. 12-18.
- [4] Putri, M.P. dan Effendi, H., 2018. Implementasi Metode Rapid Application Development Pada Website Service Guide "Waterfall Tour South Sumatera". *Jurnal SISFOKOM*, 7(2), pp. 130-136.
- [5] Aini, N., Wicaksono, S.A., dan Arwani, I., 2019. Pembangunan Sistem Informasi Perpustakaan

- Berbasis Web menggunakan Metode Rapid Application Development (RAD) (Studi pada : SMK Negeri 11 Malang). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3(9), pp. 8647-8655.
- [6] Sagala, J.R., 2018. Model Rapid Application Development (RAD) Dalam Pengembangan Sistem Informasi Penjadwalan Belajar Mengajar. *Jurnal Mantik Penusa*, 2(1), pp. 87-90.
- [7] Sasmito, G.W. dan Wiyono, S., 2017. Implementation of Rapid Application Development Method on Academic Staff System of Harapan Bersama Polytechnic. *International Journal of Computer Trends and Technology*, 50(1), pp. 11-13.
- [8] Handayani, I., Aini, Q. dan Oktavyanti, Y., 2015. Penggunaan Rinfocal Sebagai Aplikasi Pengingat (Reminder) Kegiatan Akademik Pada Perguruan Tinggi. *Creative Communication and Innovative Technology Journal*, 9(1) pp. 13-26
- [9] Reedy, K.J. dan Chary, M.V., 2013, An RFID-Based Reminder System for Smart Home. *International Journal of Scientific Engineering and Technology Research*, 02(05), pp. 318-324.
- [10] Schwebel, F.J. dan Larimer, M.E., 2018. Using text message reminders in health care services: A narrative literature review. *Internet Interventions*, 13, pp. 82-104
- [11] Palacios-Barahona, U. et al., 2018. Effectiveness of text messages, calls and e-mail on adherence to medical appointments. Review of systematic reviews and meta-analysis. *Rev CES Medicina*, 32(1), pp. 14-22.
- [12] Mekonnen, Z.A., Hussien, F.N., Tilahun, B., Gelaye, K.A., dan Mamuye, A., 2019. Development of automated text-message reminder system to improve uptake of child vaccination in Ethiopia. *Open Journal of Public Health Informatics*, 11(2).
- [13] Pop, D-P. Dan Altar, A., 2014. Designing an MVC Model for Rapid Web Application Development, *Procedia Engineering*, 64, pp. 1172-1179.
- [14] Krasner, G.E. dan Pope, S., 1988, *Journal of Object-Oriented Programming*, 1(3), pp. 26-49
- [15] Hadtanto, E. T., Budiman, A., dan Triono, J., 2016. Sistem Informasi Pemesanan Tiket Travel Berbasis GPS. *Pilar Teknologi*, 1(2), pp. 88-96
- [16] Suryasari, Callista, A., & Sari, J. (2012). Rancangan Aplikasi Customer Service Pada PT. Lancar Makmur Bersama. *JSI: Jurnal Sistem Informasi*, 4(2), pp. 468-476.
- [17] Ismael, 2017. Rancang Bangun Sistem Informasi Penyaluran Semen Padang Untuk Daerah Bengkulu Selatan Di Cv. Mutiara Bersaudara. *Jurnal EdikInformatika*, 2(2), pp. 147- 156.
- [18] Sandy dan I Dewa., 2015. Penerapan Rapid Application Development pada Sistem Penjualan Sepeda Online. *Jurnal SIMETRIS*, 6(1), pp. 27-36