IMPLEMENTASI CASHLESS PAYMENT PADA PEMBAYARAN DENDA KETERLAMBATAN PENGEMBALIAN BUKU OPEN LIBRARY UNIVERSITAS TELKOM
IMPLEMENTATION OF CASHLESS PAYMENT FOR LATE RETURN BOOKS IN OPEN LIBRARY TELKOM UNIVERSITY

Tiara[1], Nyoman Bogi Aditya Karna[2], Ratna Mayasari, S.T, M.T,[3]
[1,2,3]Prodi S1 Teknik Telekomunikasi, Fakultas Teknik, Universitas Telkom
[1]**tiaran@student.telkomuniversity.ac.id**, [2]**nyomanbogi@gmail.com**,
[3]**ratnamayasari@telkomuniversity.ac.id**

Abstrak

Cashless Payment **merupakan suatu metode pembayaran non-tunai. Metode pembayaran tersebut sudah diterapkan pada aplikasi perdangan elektronik (*e-commerce*). Dalam aplikasi e-commerce dapat menerapkan cashless payment dengan integrasi *payment gateway*. *Payment Gateway* adalah suatu jalur pembayaran non tunai yang dimana melakukan transaksi secara online dalam aplikasi. Selain aplikasi *e-commerce*, *payment gateway* dapat diintegrasikan pada aplikasi pembayaran lainnya seperti pembayaran keterlambatan pengembalian buku di Open Library Telkom University. Proses pembayaran denda saat ini masih memilki kendala yaitu peminjam tidak bisa melakukan secara *cash* melainkan *cashless*. Dengan aplikasi yang sudah terintegrasi *payment gateway*, peminjam dapat melakukan bayar denda secara mandiri atau disebut dengan *self payment*. *Self payment* tersebut dilakukan tanpa kontak secara langsung dengan admin. Sebagai konfirmasi ketika sudah selesai melakukan pembayaran dengan dikirimkan *email* kepada *user* melalui *Simple Mail Transfer Protocol* (SMTP), selain itu transaksi yang sudah dilakukan oleh *user* akan tersimpan dalam *database* sebagai riwayat pembayaran. Aplikasi pembayaran keterlambatan buku dilakukan pengujian nilai *delay* dari *client* menuju *server* pada payment gateway dan utillization resource dalam penggunaan *CPU*, *Memory* dan *Network*. Pengujian aplikasi terhadap *delay* menghasilkan 15.43 ms pada satu *device* dan 101.29 ms pada dua *device*, nilai pengujian sudah memenuhi ITU-T G.1010 dalam proses transaksi. Sedangkan *utillization resource* CPU sebesar 7%, *Memory* 10% dan *Network I/O* 151.5 Kbps**

Kata kunci : ***Cashless Payment, Payment Gateway, SMTP, Open Library, Delay, Utillization Resource***

Abstract

Cashless Payment is a non-cash payment method. This payment method has been applied to electronic trading applications (e-commerce). In e-commerce applications, cashless payments can be implemented with payment gateway integration. Payment Gateway is a non-cash payment channel which make online transactions in the application. In addition to e-commerce applications, payment gateway can be integrated into other payment applications such as late payment for returning books at the Telkom University Open Library.The process of paying fines still have obstacles, when patron can't transaction via cash, but cashless. With an application that has an integrated payment gateway, patrons can pay fines independently , which is known as self payment. Self payment not made directly interacting with admin. As confirmation when the payment has been completed by sending an email to the user via the Simple Mail Transfer Protocol (SMTP), and transactions that have been made by the user will be stored in the database as payment history.The book delay payment application has been testing for delay from the client to the server on the payment gateway and resource utilization in CPU, Memory and Network usage. Application testing for delay produces is 15.43 ms on one device and 101.29 ms on two device the value of testing has met the ITU-T G.1010 standard of delay in process transactions. Meanwhile value of utilization resource for CPU is 7%, Memory 10% and Network I/O 151.5 Kbps.

Keywords: ***Cashless Payment, Payment Gateway, SMTP, Open Library, Delay, Utillization Resource***

1. Pendahuluan

Tahun 2019, Open Library menerapkan sistem peminjaman buku *self loan, self return* dan *self pickup* dalam suatu alat yaitu *Self Service System. Self Service* merupakan alat yang terhubung dengan *Library Management System* (LMS). Alat ini menerapkan tiga sistem yaitu *self loan* peminjaman secara mandiri tanpa dengan petugas,*self return* pengembalian buku secara mandiri tidak melewati batas waktu, dan *self pickup* peminjam melakukan pemilihan buku untuk dipinjam lalu mengembalikannya secara mandiri [1].*Self return* tidak dapat menangani keterlambatan buku bersamaan dalam pembayaran denda. Keterlambatan dalam pengembalian buku, peminjam diwajibkan membayar denda tersebut secara tunai.

Cashless Payment adalah sebuah metode transaksi non-tunai melalui media seperti Kartu ATM,Kartu Debit dan Kartu Kredit, serta *virtual account* dan E-Money [2]. *Cashless payment* dapat diterapkan pada pembayaran denda keterlambatan pengembalian buku. Aplikasi pembayaran denda metode *cashless payment* dapat diterapkan dengan integrasi *payment gateway*. Aplikasi pembayaran denda akan terhubung dengan database dalam menyimpan riwayat pembayaran. Setelah melakukan transaksi, user akan menerima *email* berisikan status dengan barcode tertentu. Proses pengiriman *email* menggunakan *Simple Mail Transfer Protocol* (SMTP).

Tugas Akhir ini, berdasarkan latar belakang akan dilakukan pengembangan penelitian sebelumnya dalam *Self Service System* dengan menambahkan pada proses pembayaran denda keterlambatan pengembalian buku dengan metode *cashless payment* yang terhubung melalui *payment gateway*. Pengembangan sistem yang sudah diimplementasikan akan diberi tambahan saat pembayaran denda keterlambatan pengembalian buku secara *cashless payment*. Dengan menerapkan metode *cashless* diharapkan dapat meningkatkan efesiensi dalam proses pembayaran denda sekaligus rekap setoran secara otomatis dari pihak Open Library kepada bagian Keuangan Universitas Telkom.

1.2 Tujuan Penelitian

Bagian ini menjelaskan tujuan dari penelitian yang dilakukan. Manfaat dari perangkat tersebut diharapkan dapat dipakai guna meningkatkan efisiensi waktu dan produktivitas.

- Menerapkan sistem *cashless* pada saat pembayaran denda keterlambatan buku di Open Library Universitas Telkom yang terintegrasi dengan aplikasi
- Melakukan analisa kinerja aplikasi dengan mengukur *delay* dan *utillization resource*.

1.3 Identifikasi Masalah

- Belum diterapkan sistem cashless payment pada transaksi keterlambatan pengembalian buku.
- Keakuratan sebuah aplikasi dalam mengirimkan data dari client dan server.

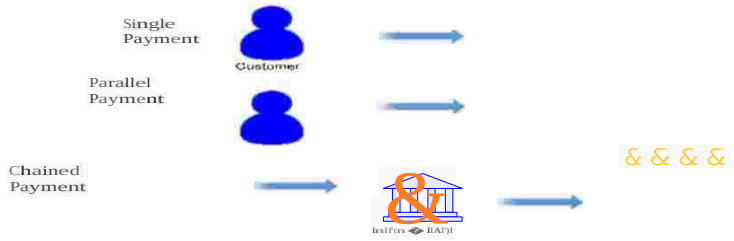
2. Material dan Metodologi

2.1 *Cashless Payment*

Cashless payment adalah sistem pembayaran secara non-tunai, artinya saat proses transaksi tidak mengeluarkan uang secara tunai melainkan melalui sebuah media.Metode ini merupakan sebuah inovasi teknologi dalam proses transaksi agar cepat dan aman tanpa mengeluarkan uang tunai.Konsep *cashless payment* selain memfasilitasi transaksi dalam sistem perokonomian menjadi transparan [3]. Salah satu penerapan dari *cashless payment* adalah *e-money*. *E-money* adalah representasi dari kartu debit dan kredit untuk digunakan dalam membeli barang selain uang kertas [4].Untuk jaman sekarang *e-money* tidak hanya melalui kartu debit dan kredit namun dapat dilakukan dengan menggunakan *mobile phone* atau disebut dengan *mobile payment*.*Mobile payment* bisa dalam bentuk *internet banking*,dan aplikasi *non bank* sebagai pihak ketiga yang dimana melakukan aktivitas dari dimulai,disahkan dan dikonfirmasi oleh *mobile phone* [5]. *Mobile payment* memiliki 3 tipe skenario yaitu:

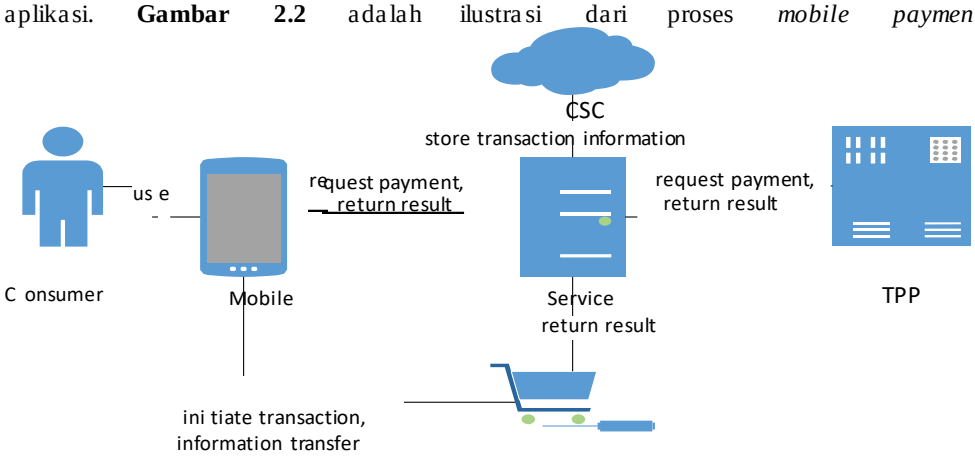
- *Simple Payment*, pembayar akan melakukan transaksi ke pedagang secara mandiri.Contohnya seseorang melakukan transaksi langsung ke konter pembayaran,hubungan ini disebut dengan 1:1 yang dimana hampir serupa dengan transaksi tradisional dijelaskan pada **Gambar 2.1(a)** [6].
- *Parallel Payment*, seorang pelanggan melakukan transaksi ke beberapa pedagang.Hubungan seperti antara pelanggan dan pedagang yaitu 1:m yang dimana dari 1 ke beberapa tujuan dijelaskan pada **Gambar 2.1(b)** [6].
- *Chain Payment*, pelanggan melakukan transaksi yang dibagi di antara beberapa pedagang, tetapi memiliki koneksi perantara antara pelanggan dan pedagang untuk melakukan itu.

Contohnya pelanggan melakukan transaksi tagihan telepon rumah, listrik dan air melalui perantara atau media sekaligus akan terhubung dengan beberapa pedagang secara otomatis. Hubungan seperti ini 1:1:m yang dijelaskan pada **Gambar 2.1(c)** [6].



Gambar 2 1 Tipe dalam transaksi mobile payment[6]

Sistem berikut mencakup Server yang menyediakan layanan sistem kepada pelanggan dan pedagang seperti pendaftaran, otentikasi identitas dan pengembalian hasil transaksi. Pelanggan yang melakukan konfirmasi pembayaran kepada pedagang maupun layanan. Pedagang menyediakan layanan komoditas, informasi transaksi dan objek kepada pelanggan. Selama inisialisasi sistem, KGC (*Key Generation Center*) menghasilkan parameter public sistem dan kunci utama sistem. *Cloud Storage Center* (CSC) digunakan untuk menyimpan informasi transaksi yang telah dienkripsi oleh pelanggan. *Third Party Payment* (TPP) yaitu pihak ketiga atau perantara seperti aplikasi. **Gambar 2.2** adalah ilustrasi dari proses *mobile payment*. [7]

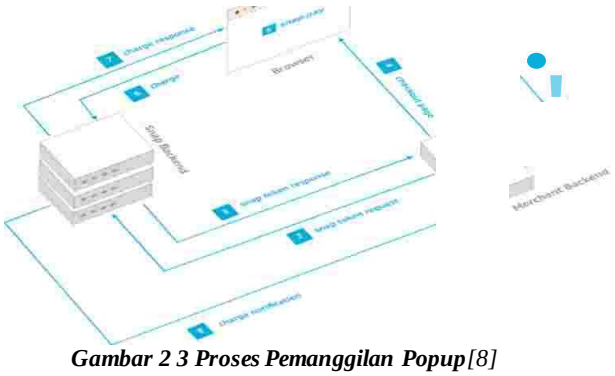


Gambar 2 2 Proses Mobile Payment[7]

Proses *mobile payment* tersebut didukung oleh *payment gateway*. *Payment Gateway* adalah layanan menyimpan informasi pembayaran dalam metode pembayaran tertentu (rincian pembayaran) dari pelanggan setelah selesai melakukan pembayaran. Informasi pesan tersebut menjelaskan bahwa bank atau *finance institution* dapat memproses lalu menerima atau menolak sebelum menyelesaikan pembayaran.

Payment Gateway sudah diterapkan di Indonesia sejak tahun 2013. Salah satu contoh payment gateway adalah Midtrans, yang dimana perusahaan tersebut berperan sebagai TPP dengan beberapa bank atau finance institution. Dalam proses integrasi midtrans untuk pemanggilan snap.js atau popup pembayaran. Pada **Gambar 2.3** menjelaskan bagaimana proses pembayaran melalui snap.js yang sudah terintegrasi dengan aplikasi kita, sebagai berikut:

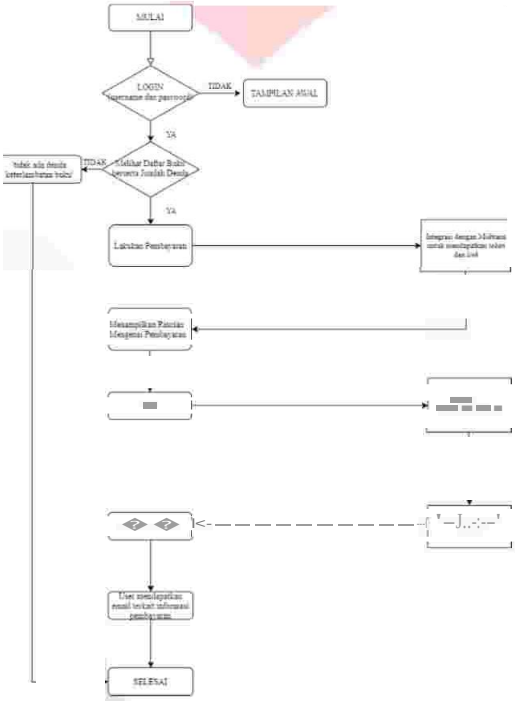
1. User melakukan pembayaran
2. Server Merchant melakukan API_request ke server Snap berisikan data transaksi
3. Server Snap menyimpan data transaksi dan memberikan SNAP_TOKEN
4. Server Merchant menampilkan halaman ringkasan pembelian dengan SNAP_TOKEN di browser
5. User memverifikasi rincian pembelian dan mengklik tombol bayar. Merchant javascript mengeksekusi snap.js. Pengguna kemudian mengisi data pembayaran dan mengklik tombol konfirmasi.
6. Snap JS mengirimkan data pembayaran ke server Snap
7. Server Snap memproses data dan meresponse dengan status pembayaran. Snap JS kemudian mengeksekusi callback sesuai dengan status transaksi
8. Server Snap akan memberikan notifikasi ke server Merchant tentang status pembayaran [8]



Gambar 2 3 Proses Pemanggilan Popup*[8]*

2.2 Cara Kerja Aplikasi

Payment gateway yang sudah terhubung dengan Aplikasi akan dilakukan ujicoba transaksi dengan *sandbox* dari pihak MIDTRANS, sebelum melakukan ujicoba tersebut berikut cara kerja pada Aplikasi Pembayaran Denda Keterlambatan Buku.



Gambar 2.4 Flowchart Aplikasi

Tahap 1 : User login dengan menggunakan akun SSO Telkom
Tahap 2 : Jika User berhasil login maka akan menampilkan data terkait denda pengembalian buku. Jika tidak maka akan dikembalikan ke halaman utama
Tahap 3 : Menampilkan daftar denda pengembalian buku milik User, jika tidak maka proses selesai
Tahap 4 : Melakukan pembayaran terhadap denda pengembalian buku yang sudah terintegrasi dengan MIDTRANS
Tahap 5 : MIDTRANS akan membuat sebuah token terkait informasi tentang (jumlah yang dibayarkan, info buku, nama dan nim). Jika format yang dikirimkan sesuai dengan ketentuan. Maka akan disetuju untuk melanjutkan pembayaran
Tahap 6 : Menampilkan rincian pembayaran
Tahap 7 : Klik tombol bayar yang sudah terintegrasi dengan tampilan dari pihak MIDTRANS

Tahap 8 : Halaman *snap* dari MIDTRANS yang menampilkan rincian serta metode pembayaran yang akan dipilih
Tahap 9 : MIDTRANS akan memproses pembayaran sesudah memilih metode pembayaran beserta akun User
Tahap 10 : Setelah memproses pembayaran MIDTRANS akan memberikan *HTTP response* terkait status pembayaran
Tahap 11 : *Response* akan dikirimkan menuju *link* yang sudah diintegrasikan pada akun MIDTRANS. *link* tersebut merupakan halaman notifikasi pembayaran.
Tahap 12 : Setelah mendapat notifikasi pembayaran, user akan mendapatkan email terkait status pembayaran seperti yang ada di notifikasi.
Tahap 13 : Selesai

2.1 Spesifikasi Perangkat

Untuk pengujian *delay* menggunakan *wireshark* dan untuk *utillization resource* menggunakan *resource monitoring* OS Windows. Berikut spesifikasi perangkat keras dalam mendukung proses pengujian.

Tabel 2 1 Spesifikasi Perangkat Keras

Kompenen Laptop	
Processor	Intel(R) Core (TM) i5-6200U CPU @ 2.30GHz (4 CPUs), ~2.4GHz
Memory	12288MB RAM

2.4 Pengujian

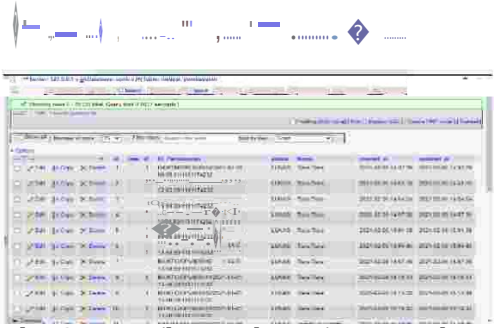
Pengujian yanag akan dilakukan pada Tugas Akhir ini terbagi dalam beberapa tahapan. Tahapan-tahaan tersebut adalah sebagai berikut:

1. Pengujian *Cashless Payment* dan *email*
   a. Pada pengujian *cashless payment* dengan melakukan uji coba 20 transaksi keterlambatan dengan data buku
   b. Data buku tersebut memiliki *barcode* atau ID_Pembayaran dengan format "*Barcode buku/Tanggal Pengembalian/NIM* ".
   c. Pengujian riwayat pembayaran yang sudah terintegrasi dengan ID_Pembayaran dan status.
   d. Pada pengujian *email* memastikan konfirmasi terkait pembayaran diterima oleh *user*.
2. Pengujian *Delay* dan *Utillization Resource*
   a. Pada pengujian *delay* akan dilakukan membaca trafik jaringan dengan aplikasi *Wireshark* pada saat proses transaksi dari memilih buku sampai dengan *user* memperoleh *email* status pembayaran.
   b. Pengujian *utillization resource* akan dilakukan dengan *resource monitor* yang dimiliki oleh sistem operasi Windows. Waktu monitoring dimulai bersamaan dengan pada saat pengujian *delay*.

3. Pembahasan

3.1. Hasil dan Analisis Pengujian pada *sandbox*

Uji coba 20 transaksi pada buku yang berbeda dan *user* berbeda, bahwa status yang diperoleh dari *server* MIDTRANS yaitu "200 == LUNAS" berhasil tersimpan yang dimana ketika *user* melakukan *login* kembali hanya transaksi yang belum dilakukan. Untuk pengiriman *email* pada aplikasi menuju *user* terkait konfirmasi pembayaran berhasil bersamaan dengan tersimpannya pada riwayat pembayaran.



Gambar 3.1 Tampilan Database Riwayat Pembayaran

Gambar 3.2 Tampilan penerimaan *email*

3.2 Hasil dan Analisis Pengujian *Delay* antar Aplikasi dengan *Payment Gateway*

Tabel 3.1 Hasil Pengukuran Delay

Percobaan ke-	*Delay (s)*
1	***0.08199***
2	***0.01833***
3	***0.0106***
4	***0.00749***
5	***0.01151***
6	***0.01213***
7	***0.00764***
8	***0.01904***
9	***0.00724***
10	***0.00677***
11	***0.01285***
12	***0.01272***
13	***0.01209***
14	***0.01025***
15	***0.01229***
16	***0.01605***
17	***0.01344***
18	***0.01373***
19	***0.0096***
20	***0.01292***

Hasil rata-rata delay yang diperoleh berdasarkan **Tabel 3.1** bahwa untuk aplikasi pembayaran denda keterlambatan buku memiliki rentang rata-rata delay 0.00677 – 0.08199 s atau 6.77 - 81.99 ms. Pengaruh nilai delay terhadap kecepatan dalam proses pengiriman data dari client dan server adalah semakin besar sebuah nilai delay seperti percobaan pertama maka waktu proses pengiriman data antar client dan server semakin lama.

3.3 Hasil dan Analisi Pengujian *Delay* antar Aplikasi Client dan Aplikasi Server

Berikut adalah hasil pengujian *delay* antar aplikasi *client* dan aplikasi *server* dengan *device* yang berbeda.

Tabel 3 1 Hasil Pengujian Delay 2

Percobaan ke-	*Delay (s)*
1	***0.0819917***
2	***0.1060231***
3	***0.0859842***
4	***0.1091612***
5	***0.0952623***
6	***0.0587871***
7	***0.0790361***
8	***0.0832137***
9	***0.1416908***

10	***0.0999401***
11	***0.0867081***
12	***0.1072425***
13	***0.0802171***
14	***0.1395067***
15	***0.1033852***
16	***0.1576176***
17	***0.1006606***
18	***0.0933982***
19	***0.1014089***
20	***0.1147613***

Menurut hasil pengujian *delay* antar aplikasi *client* dan *server* yang tertera pada **Tabel 3.2** dengan rentang nilai yang diperoleh 0.0587871 – 0.1576176 s. Pengukuran *delay* pada 20 percobaan memiliki rata-rata sebesar 0.1012998 s dengan nilai deviasi sebesar 0.0105639 s. Kenaikan nilai *delay* tertinggi pada saat percobaan ke -9 sebesar 0.0584771 s. Penurunan nilai *delay* terjadi pada saat percobaan ke-17 sebesar 0.056957 s. Percobaan ke 1 sampai dengan percobaan ke 5 bersifat akurat dan presisi nilai pada setiap percobaan sama dengan *nilai rata-rata delay ± nilai deviasi*. Selanjutnya percobaan ke-6 sampai percobaan ke-16 mengalami kenaikan dan penurunan yang tidak bersifat akurat dan presisi. Percobaan ke-17 sampai percobaan ke-20 memiliki nilai yang bersifat akurat dan presisi.

Berdasarkan nilai *delay* yang diperoleh dari rentang 0.0587871 – 0.1576176 s atau 58.78 – 157.61 ms dengan rata-rata 0.1012998 s atau 101.29 ms bahwa nilai tersebut sudah memenuhi standar ITU-T G.1010 dalam mengukur nilai *delay* untuk proses transaksi sebesar lebih kecil dari 2 s.

3.4 Hasil dan Analisis Pengujian pada *utillization resource*

Berikut adalah hasil pengujian terhadap *utillization* untuk mengetahui konsumsi CPU, Memory dan Network.

Tabel 3.3 Hasil Pengujian *Utillization Resource*

Kondisi Akhir				
CPU Utilization	**CPU Usage**	**Memory Usage**	**Network Utilization**	**Network I/O (Kbps)**
8%	**2%**	**56%**	**0%**	**3**
53%	**9%**	**58%**	**0%**	**77**
20%	**3%**	**56%**	**3%**	**21**
29%	**4%**	**58%**	**0%**	**24**
31%	**9%**	**58%**	**0%**	**52**
64%	**7%**	**57%**	**0%**	**21**
56%	**6%**	**57%**	**0%**	**91**
72%	**12%**	**53%**	**0%**	**243**
74%	**9%**	**55%**	**0%**	**108**
54%	**8%**	**53%**	**0%**	**221**
53%	**7%**	**50%**	**0%**	**223**
52%	**7%**	**50%**	**0%**	**218**
51%	**9%**	**49%**	**0%**	**223**
48%	**7%**	**48%**	**0%**	**224**
47%	**6%**	**49%**	**0%**	**218**
44%	**7%**	**49%**	**0%**	**221**
40%	**5%**	**49%**	**0%**	**221**
43%	**5%**	**50%**	**0%**	**200**
41%	**6%**	**50%**	**0%**	**200**
41%	**6%**	**51%**	**0%**	**222**

Berdasarkan **Tabel 3.3** dalam mengukur *utillzation resource* pada Aplikasi Pembayaran Keterlambatan Buku, pada saat kenaikan nilai *utilization CPU* maka nilai *Usage CPU* akan meningkat namun pada saat *Usage CPU* meningkat nilai *utilization CPU* belum tentu akan meningkat. Nilai *utillization CPU* meningkat disebabkan aktivitas pada keseluruhan aplikasi dalam perangkat meskipun aplikasi lain tidak dioperasikan, sedangkan *Usage CPU* pada aplikasi pembayaran keterlambatan buku yang ditinjau melalui *ngrok* meningkat disebabkan ada activitas pada aplikasi tersebut.

Untuk *usage CPU* pada aplikasi pembayaran keterlambatan buku memiliki rentang nilai 2-12% dan rata-rata 7%. Sedangkan utilitas pada *Memory* memiliki rentang 48-58% dengan rata-rata 53% dan penggunaan sebesar 10%. Untuk utilitas pada *Network* memiliki kecepatan rata-rata sebesar 151.5 Kbps dengan packet loss 0%.

4. Kesimpulan dan Saran

4.1. Kesimpulan

Dari penelitian yang telah dilakukan dapat diambil keisimpulkan bahwa telah berhasil dibuat aplikasi "Implementasi Cashless Payment pada Pembayaran Denda Keterlambatan Buku" menggunakan *payment gateway* dengan rincian penelitian sebagai berikut:

1. Aplikasi Pembayaran Keterlambatan Buku dapat beroperasi dengan lancar, dan rekapan tersimpan dalam riwayat pembayaran. Status "LUNAS" didapatkan ketika mendapat response 200 dari server Midtrans.
2. Berdasarkan nilai delay yang diperoleh untuk satu device dari rentang dari 6.77 - 81.99 ms dengan rata-rata 15.43 ms. Sedangkan uji coba delay untuk 2 device berbeda memiliki rentang 0.0587871 – 0.1576176 s atau 58.78 – 157.61 ms dengan rata-rata 0.1012998s atau 101.29 ms bahwa nilai tersebut sudah memenuhi standar ITU-T G.1010 dalam mengukur nilai delay untuk proses transaksi sebesar lebih kecil dari 2 s.
3. Utilization Resource pada aplikasi membutuhkan konsumsi 7% CPU, 10% Memory dan kecepatan dalam pengiriman data dibutuhkan 151.5 Kbps.

4.2. Saran

Adapun saran untuk pengembangan tugas akhir ini sebagai berikut:

1. Penggunaan kualitas perangkat keras pada processor, seperti kualitas CPU lebih baik dari sebelumnya
2. Penggunaan bahasa pemrograman yang dalam pengiriman data cepat, seperti NodeJs.
3. Pengujian *delay* untuk beberapa *user* dalam proses transaksi pada website yang tidak berbasis *localhost*

Referensi

[1] N. Karna, D. Pratama, and M. Ramzani, "Self Service System for Library Automation Case Study at Telkom University Open Library," *2019 Int. Conf. Inf. Commun. Technol.*, pp. 689–693, 2019.

[2] H. Kartika, "Secure Cashless Payment Governance in Indonesia : A Systematic Literature Review," *2018 Int. Conf. ICT Smart Soc.*, pp. 1–4, 2018.

[3]F. A. S. Khan and S. M. Damanhouri, "Cashless Payment Vs Conventional Payment System –," vol. 29, no. 3, pp. 613–620, 2017.

[4] C. Patil and K. Hittinhalli, "E-CASH : A Novel Approach of Virtual Money Transaction Using Smart Cards," no. 5, pp. 157–161, 2017.

[5] D. Tamara, F. Roesmawi, H. Febria, and I. D. Ariesta, "Customer Loyalty Indicator of Mobile Payment Application in the Financial Service Industry : A Study of LinkAja," 2020, vol. 08, no. 01, pp. 1527–1539, doi: 10.18535/ijsrm/v8i01.em05.

[6] P. Pukkasenung, "An Efficient of Secure Mobile Phone Application for Multiple Bill Payments," pp. 4–9, 2016, doi: 10.1109/WAINA.2016.63.

[7] X. Zhang and H. Zeng, "Mobile Payment Protocol Based on Dynamic Mobile Phone Token," 2017.

[8] Midtrans, "API documentation," 2017.