

Rancang Bangun API Point of Sales Terintegrasi dengan Thermal Printer Menggunakan MQTT Gateway

Designing Point of Sales API Integrated with Thermal Printer Using MQTT Gateway

Mohamad Fahmi Syaifudin, Nanang Fakhru Rozi

Jurusan Teknik Informatika, Institut Teknologi Adhi Tama Surabaya

Jl. Arief Rahman Hakim No. 100, Sukolilo, Surabaya

fahmisyiaifudin00@gmail.com,

Abstract— A point of sale is a place where customers pay to buy a product. Nowadays, there are many platforms providing services for mobile or desktop-based points of sale. However, web-based point of sale is seldom used as it does not support interconnection with thermal printer hardware for printing receipts. Therefore, this study aims to design and develop a point-of-sale Application Programming Interface (API) that is integrated with thermal printer. By using this system, business owners are expected to be able to monitor their sales transactions and print transaction receipts when an order comes in and easily integrate the existing applications with thermal printers, including those with web, Android, or desktop bases. The integration process utilizes the payload data of transactions sent to the API Server. After that, it sent the payload data to the printer driver using the Message Queuing Telemetry Transport (MQTT) protocol. This research employed the method of software development known as Agile Extreme Programming (XP). The development model consists of the planning, design, coding, and testing processes. The researcher involved integration testing, payload data testing, API performance testing, and user acceptance testing. After conducting those tests, the results indicated that the built-in integrator could be integrated with the web, WordPress, Android, and desktop. It could send the payload data up to 5 MB. The API Server could handle requests from up to 1000 users simultaneously with an average latency of 11.58 ms. The results of user acceptance testing met the user's needs and the respondents could understand it easily as it got an average score of 92.68% (strongly agree) based on the Likert scale calculation.

Keywords : Point of sales, Thermal Printer, POS API, MQTT, Agile Extreme Programming

Abstrak— Point of sales adalah tempat pembayaran para pelanggan dalam membeli sebuah produk. Saat ini banyak beredar di pasaran, platform penyedia layanan point of sales berbasis mobile atau desktop, sedangkan point of sales berbasis web sangat jarang digunakan karena tidak mendukung interkoneksi dengan hardware thermal printer untuk mencetak struk. Penelitian ini bertujuan untuk merancang dan membangun Application Programming Interface (API) point of sales yang terintegrasi dengan thermal printer. Harapannya para pemilik usaha dapat memonitor transaksi penjualannya dan mencetak struk transaksi ketika ada pesanan yang masuk dan memudahkan dalam mengintegrasikan aplikasi yang sudah ada dengan thermal printer, baik aplikasi berbasis web, Android, maupun desktop. Proses integrasi yang dimaksud adalah dengan memanfaatkan payload data transaksi yang dikirim ke API Server, kemudian mengirimkan payload data tersebut kepada driver printer menggunakan protokol Message Queuing Telemetry Transport (MQTT). Penelitian ini menggunakan metode pengembangan perangkat lunak Model Agile extreme programming (XP). Model pengembangan tersebut terdiri dari

proses *planning, design, coding* dan *testing*. Pengujian dari penelitian ini terdiri dari *integration testing, payload data testing, API performance testing*, dan *user acceptance testing*. Hasil pengujian menunjukkan bahwa integrator yang dibangun dapat diintegrasikan dengan web, Wordpress, Android, dan desktop. Payload data yang dapat dikirim hingga 5MB. API Server dapat meng-handle request hingga 1000 user secara bersamaan dengan rata-rata latency 11,58ms. Hasil user acceptance testing telah memenuhi kebutuhan pengguna dan mudah dipahami oleh responden dengan mendapatkan rata-rata skor 92,68% (sangat setuju) berdasarkan perhitungan skala likert

Kata Kunci : Point of sales, Thermal Printer, POS API, MQTT, Agile Extreme Programming

I. PENDAHULUAN

Saat ini banyak beredar di pasaran platform penyedia layanan point of sales beserta printer-nya untuk mencetak struk transaksi jual beli oleh pelaku usaha. Platform tersebut menggunakan aplikasi berbasis mobile apps, atau desktop app di sisi point of sales-nya, dan menggunakan website untuk memantau dashboard penjualan, menambahkan barang, stok atau melihat report penjualan di periode tertentu. Penggunaan aplikasi berbasis mobile apps, atau desktop app di sisi point of sales bukan tanpa alasan, Hal ini disebabkan karena aplikasi berbasis website tidak mendukung interkoneksi dengan hardware printer untuk mencetak struk [2].

Pelaku usaha dapat memilih memakai platform penyedia layanan point of sales, atau mengembangkan sendiri aplikasi mereka sesuai kebutuhan. Website builder dan e-commerce merupakan solusi bagi pelaku usaha kecil dan menengah untuk membangun sistem penjualan usaha mereka, tanpa harus menyusun program dan menyewa tim pengembang dengan biaya yang mahal. Beberapa website builder dan e-commerce tidak memiliki dashboard khusus untuk pelaku usaha untuk memantau dan memonitor usahanya, dan juga website builder dan e-commerce memiliki keterbatasan interkoneksi antar-hardware printer point of sales untuk cetak struk dikarenakan beberapa website builder dan e-commerce dijalankan dalam server cloud

Berdasarkan permasalahan tersebut, penulis mendapatkan ide untuk mengembangkan API Point of Sales yang terintegrasi dengan Thermal Printer. Rancangan sistem tersebut dapat dijalankan dengan memanfaatkan API (Application Programming Interface) yang disediakan. Sehingga memudahkan tim pengembang

dan pemilik usaha dalam mengintegrasikan aplikasi yang sudah ada dengan *printer Point of Sales*, baik aplikasi berbasis web, Android, maupun desktop.

II. DASAR TEORI

A. Point of Sales

Point of sales atau POS adalah tempat pembayaran para pelanggan dalam membeli sebuah produk. Biasanya POS berbentuk mesin, tablet, smartphone, mesin EDC atau perangkat lainnya yang digunakan untuk transaksi di toko seperti pada gambar 1. Manfaat POS bagi pelaku usaha adalah menyimpan harga barang, mencatat pendapatan pengeluaran dan laba, mencatat secara otomatis barang yang masuk dan keluar, dan menghemat budget dalam merekrut tim khusus [4].



Gambar 1. Point of Sales [4]

B. REST API

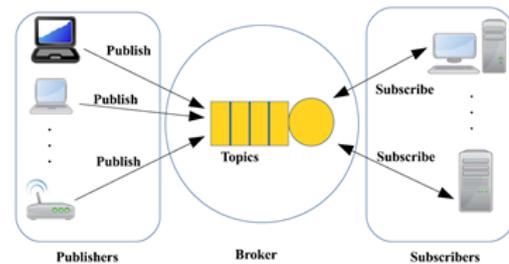
REST API (*Representational State Transfer Application Programming Interface*) merupakan standar dari arsitektur untuk pengembangan sistem berbasis web sebagai protokol untuk komunikasi data. Arsitektur REST merupakan hubungan antara client dan server, yang dimana sumber daya/data disediakan oleh REST server dan sumber daya/data tersebut untuk selanjutnya diakses dan ditampilkan oleh REST client. Setiap sumber daya/data diidentifikasi oleh URL (*Universal Resource Identifiers*) [5]. Setiap sumber daya/data tersebut kemudian ditampilkan dalam bentuk teks, XML atau JSON. Umumnya menggunakan XML dan JSON. REST menjelaskan bagaimana aplikasi berkomunikasi melalui *Hypertext Transfer Protocol* (HTTP).

C. MQTT

MQTT (*Message Queuing Telemetry Transport*) adalah protokol pengiriman pesan yang diperkenalkan oleh Andy Stanford-Clark dari IBM dan Arlen Nipper dari Arcom di tahun 1999 dan telah distandardisasi pada 2013 di OASIS. MQTT bertujuan menghubungkan perangkat *publisher* dan *subscriber* menggunakan aplikasi dan *middleware*. Perangkat terkoneksi menggunakan mekanisme *routing* (*one-to-one, one-to-many, many-to-many*). MQTT biasanya digunakan untuk menghubungkan protokol untuk *Internet of Things* (IoT) dan *Machine to Machine* (M2M) [1].

MQTT menggunakan prinsip *publish subscribe* seperti yang pada Gambar 2. *Publisher* mengirim *topics* yang berisi data ke *broker*, kemudian *subscriber* meminta

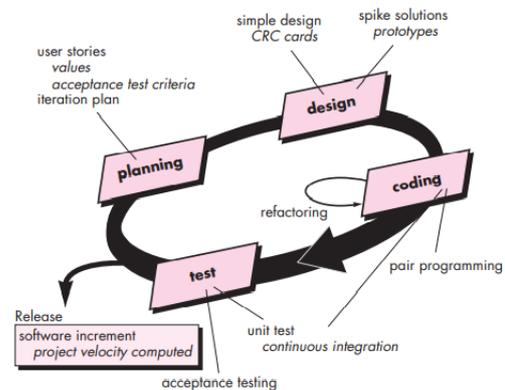
topics yang dikirim *publisher*, sehingga *publisher* dan *subscriber* dapat berkomunikasi



Gambar 2. Arsitektur MQTT [1]

III. METODE

Metode yang digunakan dalam Menyusun siklus pengembangan aplikasi adalah dengan metode *agile extreme programming*. Metode *agile extreme programming* adalah metodologi pengembangan perangkat lunak tangkas yang menangani perubahan yang cepat dalam persyaratan dan lingkungan secara bertahap dan siklus rilis [7]. Tahapan-tahapan yang harus dilalui antara lain: *Planning, Design, Coding, dan Testing* [8] seperti tampak pada gambar 2.



Gambar 2. Metode Agile Extreme Programming [8]

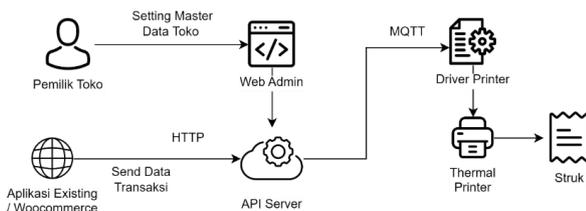
IV. HASIL DAN PEMBAHASAN

A. Planning

Perencanaan terhadap software yang diinginkan mengacu pada *user stories*. *User stories* menggambarkan fitur dan fungsi yang dibutuhkan terhadap software tersebut. Ketika semua *user stories* telah ditentukan, *developer* akan menentukan lama pengerjaan untuk tiap-tiap *user stories*. Adapun beberapa fungsi dan kebutuhan dari aplikasi *Point of Sales* sebagai berikut

- 1) Mencetak struk transaksi penjualan melalui aplikasi yang sudah ada
- 2) Membuat halaman untuk mengatur informasi pada struk, seperti nama toko, alamat dan nomor telepon
- 3) Membuat *dashboard* monitoring dan analisis penjualan selama periode waktu tertentu dari data transaksi

B. Desain



1) Desain Sistem

Gambar 3. Perancangan Sistem

Pada gambar 3 perencanaan Aplikasi yang sudah tersedia (*existing*) akan mengirim data transaksi dalam bentuk JSON ke API Server. API Server akan mengirim *message* ke *broker* MQTT yang nantinya akan ditangkap oleh *driver* untuk selanjutnya dicetak kedalam perangkat *thermal printer*.

Penggunaan REST API merupakan arsitektur komunikasi yang sering digunakan dalam pertukaran data antar aplikasi. Penggunaan REST API sifatnya universal artinya hampir semua Bahasa pemrograman mempunyai fitur untuk mengirim *request* API.

Penggunaan protokol MQTT bertujuan karena merupakan protokol yang populer dalam *Internet of Things*, dikarenakan *low latency*, hemat bandwidth dan hemat energi [3]. Selain itu MQTT adalah teknologi *messaging* berbasis antrean yang artinya seluruh *request* yang masuk akan diantrikan satu per satu sehingga tidak terjadi *request* yang terlewatkan.

2) Desain Aplikasi

Di sisi aplikasi dibagi menjadi tiga bagian, yang pertama adalah web admin, API Server, dan *driver*. Web admin berfungsi untuk mengatur management perangkat dan management data toko. API Service berfungsi untuk menerima *request* data transaksi yang masuk dari aplikasi *existing* lalu mengirimnya ke sisi hardware. *Driver printer* merupakan sebuah *gateway* untuk menghubungkan data yang diterima dari MQTT kemudian dikonversi menjadi perintah ESC/POS. ESC/POS adalah sebuah standard sintaks perintah yang digunakan beberapa *thermal printer*.

C. Coding

Proses coding mengimplementasikan *Framework MVC* (*Model, view, controller*), dengan menggunakan *Framework MVC* hasil *source code* dipecah menjadi beberapa bagian, sehingga aplikasi yang dikembangkan memiliki arsitektur yang selaras dan terkelola lebih baik [6].

1) API Server

API Server berfungsi untuk menerima *request* data transaksi yang masuk dari aplikasi *existing* lalu mengirimnya ke sisi hardware, Selain itu API Server ini juga memiliki fungsi untuk meng-*handle* proses logika yang ada didalam web admin mulai dari *login, register,*

dan informasi dan credential *merchant* seperti pada tabel 1.

TABEL I. API SPESIFIKATION

Method	Endpoint	Deskripsi	Body
POST	/register	Mendaftarkan <i>merchant</i> baru	{ "name": "", "phone": "", "email": "", "password": "", "repeat_password": "" }
GET	/profile	Mengambil data <i>profile</i>	
POST	/print	Menambahkan dan mencetak transaksi	{ "cashier": "", "customer": "", "cash": "", "items": [{ "name": "", "qty": "", "price": "" }] }
PUT	/api/profile	Meng-update <i>profile merchant</i>	{ "name": "", "address": "", "password": "", "repeat_password": "" }
GET	/api/dashboard/summary?start=&end=	Mengambil ringkasan transaksi periode tertentu	
GET	/api/dashboard/graph?start=&end=	Mengambil grafik transaksi selama periode tertentu	
GET	/api/dashboard/top-product?start=&end=	Mengambil trend produk selama periode tertentu	
GET	/api/transaction	Mengambil seluruh data transaksi	
GET	/api/download-config	Men-download file konfigurasi <i>driver printer</i>	

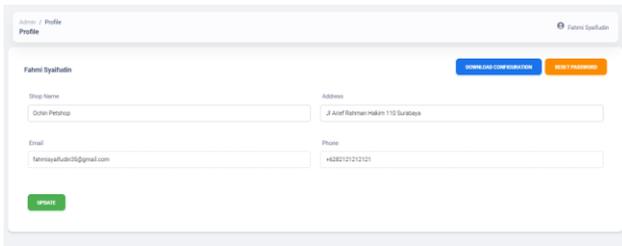
2) Web Admin

Web admin merupakan halaman web sebagai tempat *merchant* untuk melihat profile, data konfigurasi, dan memonitor transaksi yang masuk beserta visualisasinya dalam bentuk grafik seperti pada gambar 4 dan 5. Selain



fitur tersebut web admin juga berfungsi untuk mendownload file konfigurasi untuk menjalankan *driver printer*

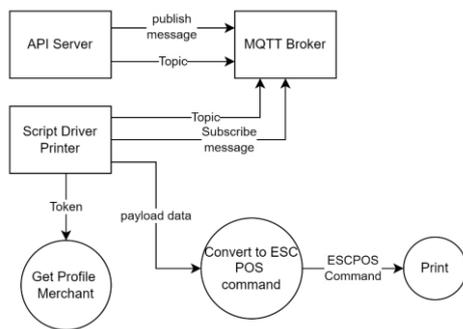
Gambar 4. Tampilan Dashboard



Gambar 5. Tampilan Profile

3) Driver Printer

Driver printer merupakan sebuah gateway untuk menghubungkan data yang diterima dari MQTT kemudian dikonversi menjadi perintah ESC/POS seperti dijelaskan pada gambar 6. ESC/POS adalah sebuah standard sintaks perintah yang digunakan beberapa thermal printer



Gambar 6. Alur kerja driver printer

D. Testing

1) Integration Testing

Tujuan integration testing dalam pengujian perangkat ini adalah memastikan integrator printer POS dapat dipakai di beberapa perangkat lunak yang akan diintegrasikan dan dapat mencetak struk dari transaksi yang diinputkan. Hasil integration testing ditunjukkan pada tabel 2.

TABEL II. INTEGRATION TESTING TERHADAP BEBERAPA PLATFORM

Platform	Metode Integrasi	Keberhasilan (Berhasil/Gagal)
Web (Codeigniter)	REST API	Berhasil
Wordpress (Woocommerce)	Webhook	Berhasil
Android (React Native)	REST API	Berhasil
Desktop (Electron)	REST API	Berhasil

2) User Acceptance Testing

User Acceptance Testing (UAT) terhadap system diperlukan untuk mengukur system yang dikembangkan sesuai dengan kebutuhan user dan kenyamanan end-user terhadap aplikasi dan system yang sudah dibuat. Tolak ukur kualitas kebergunaan aplikasi adalah ISO 9126, sehingga pertanyaan disusun berdasarkan karakteristik kualitas perangkat lunak ISO 9126. Responden dari User acceptance testing (UAT) ini adalah 16 orang para pengembang (developer) dari latar belakang perusahaan yang berbeda-beda dan dihitung menggunakan

perhitungan skala likert. Hasil user acceptance testing ditunjukkan pada tabel 3

TABEL III. USER ACCEPTANCE TESTING MENGGUNAKAN ISO 9126

Sub faktor	Deskripsi	Persentase	Penilaian
Suitability	Kemampuan perangkat lunak untuk menyediakan serangkaian fungsi yang sesuai untuk tugas-tugas tertentu dan tujuan pengguna	97,08%	Sangat Bagus
Interoperability	Kemampuan perangkat lunak untuk berinteraksi dengan satu atau lebih sistem tertentu	93,75%	Sangat Bagus
Security	Kemampuan perangkat lunak untuk mencegah akses yang tidak diinginkan	96,25%	Sangat Bagus
Understandability & Learnability	Kemampuan perangkat lunak dalam kemudahan untuk dipahami dan dipelajari	85%	Sangat Bagus
Operability	Kemampuan perangkat lunak dalam kemudahan untuk dioperasikan	82,5%	Sangat Bagus

Berdasarkan hasil keseluruhan sub faktor User Acceptance Testing, mendapatkan rata rata skor 92,68% (sangat setuju) menggunakan perhitungan skala likert

3) Payload Data Testing

Tujuan dari pengujian payload data adalah mengetahui apakah data yang dikirim sama atau tidak antara disisi pengirim dan penerima. Pengertian sama disini adalah sama dari sisi ukuran datanya dalam bentuk (byte), maupun keseluruhan karakter (string) datanya. Salah satu metode yang dapat digunakan adalah dengan menverifikasi hasil sha256 checksum disisi pengirim dan penerima. Hasil payload data testing ditunjukkan pada tabel 4

TABEL IV. PAYLOAD DATA TESTING

Jumlah Item	Data (dalam Byte)		Hasil Checksum
	Dikirim	Diterima	
5	384	384	Sesuai
10	673	673	Sesuai
20	1251	1251	Sesuai
40	2407	2407	Sesuai
60	3563	3563	Sesuai
80	4719	4719	Sesuai
100	5875	5875	Sesuai

4) API Performance Testing

Pengujian performansi API service dilakukan dengan menggunakan aplikasi JMeter. Skenario pengujiannya adalah mensimulasikan jumlah user yang mengakses dengan mendefinisikan number of threads. kemudian mengamati performansi nya dalam latency, maximum latency, dan success rate. Hasil API Performance testing ditunjukkan pada tabel 5

TABEL V. API PERFORMANCE TESTING

Jumlah User	Rasio Kesuksesan (%)	Total Latency (ms)	Total Packet (bytes)	Latency (ms)
1	100	1595	396	4,02
10	100	1340	396	3,38
20	100	1289	396	3,25
50	100	1509	396	3,80
100	100	1388	396	3,50
200	100	1343	396	3,39
500	100	3780	396	9,54
1000	100	4588	396	11,58

V. KESIMPULAN

Dari pembuatan API *Point of Sales* Terintegrasi dengan *Thermal Printer* Menggunakan MQTT Gateway ini didapat beberapa kesimpulan yaitu:

- 1) Perancangan API *Point of Sales* Terintegrasi dengan *Thermal Printer* berhasil dibangun dengan model *Agile Extreme Programming*. Proses integrasi dengan cara mengirimkan *payload* data transaksi ke API Server, kemudian dari API Server mengirimkan *payload* data tersebut kepada *driver printer* menggunakan protokol MQTT
- 2) Perancangan Integrator *thermal printer Point of Sales* berhasil diintegrasikan dengan aplikasi berbasis web. web *builder* (Woocommerce), Android dan desktop
- 3) API server yang juga berfungsi sebagai API *gateway* yang telah dibangun dapat mengirimkan hingga 5MB data *payload* transaksi dan dapat meng-*handle request* hingga 1000 user secara bersamaan dengan maksimal *latency* aktual sebesar 11,58ms
- 4) Aplikasi yang dibangun telah memenuhi kebutuhan pengguna dan mudah dipahami oleh responden

dengan mendapatkan rata rata skor 92,68% (sangat setuju) berdasarkan perhitungan skala *likert*

REFERENCES

- [1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, Protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015
- [2] A. Bjørn-Hansen, T. A. Majchrzak, and T.-M. Grønli, "Progressive web apps: The possible web-native unifier for Mobile Development," *Proceedings of the 13th International Conference on Web Information Systems and Technologies*, 2017.
- [3] A. Zainudin, M. F. Syaifudin, and N. Syahroni, "Design and implementation of node gateway with MQTT and CoAP protocol for IOT Applications," 2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE), 2019.
- [4] C. Jessica, "Point of sales: Pengertian, Jenis, Dan Fungsinya bagi bisnis," *Glints Blog*, 14-Mar-2022. [Online]. Available: <https://glints.com>. [Accessed: 14-Jul-2022].
- [5] I.G.M Ariantara, I Arwani., and W.H.N Putra. "Penerapan REST API dalam Pengembangan Aplikasi Pemesanan Rental Mobil berbasis Web dan Mobile". *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vol 4, No. 8, pp. 2569-2576, 2020.
- [6] N. F. Rozi, M. Ruswiansari, A. Rachman, S. R. Wardhana, and L. Istiyanto, "The development of lidi: A web-based car rent marketplace application in Sidoarjo, Indonesia," *IOP Conference Series: Materials Science and Engineering*, vol. 462, p. 012052, 2019.
- [7] R. Parlita, M. F. Hidayat, H. R. Putra, V. H. Satria, H. F. Lesmana, and F. H. Pralas, "Studi Komparatif model proses Perangkat Lunak TERHADAP Karakteristik Sistem Erp," *Jurnal IPTEK*, vol. 22, no. 2, pp. 1–8, 2019.
- [8] R. S. Pressman and B. R. Maxim, *Software engineering: A practitioner's approach*. New York, NY: McGraw-Hill Education, 2020.