

Pengembangan *Transaction Monitoring* Pajak Berbasis IoT Dengan Fitur *Script Auto Reboot* Studi Kasus : PT. Pinisi Elektra

Hartini¹⁾, Sri Primaini²⁾, Nurhayani³⁾, Sukro Nanda Darfero⁴⁾

^{1),2),4)} Program Studi Teknik Komputer AMIK Sigma Palembang

³⁾ Program Studi Sistem Informasi STMIK MBC Palembang

Jl Perintis Kemerdekaan No. 62 Palembang

Email : arpi.hartini.my@gmail.com¹⁾, sri.primaini@gmail.com²⁾, nurhayani141169@gmail.com³⁾, sukronando4@gmail.com⁴⁾

ABSTRACT

Online tax payments are one way to speed up the tax reporting process for taxpayers. The development of the online system is still often experiencing problems, especially the accumulation of data which eventually the online system experiences obstacles that result in the payment process being hampered. Another problem that may occur is that there are still many taxpayers who do not know and understand the online system properly. The development of transaction monitoring has a weakness where the computer system connection will die if it is used continuously to monitor transactions that occur within 1 or 2 weeks depending on the temperature conditions of the system in which it is placed. Therefore, it is necessary to add the auto reboot feature. The development of internet of things (IoT)-based tax transaction monitoring will make it easier for taxpayers to pay taxes because there is a link between taxpayers and companies engaged in tax payment services so that taxpayers are no longer bothered with the matter of paying taxes. The purpose of this study is to design an IoT-based transaction monitoring method and connect transaction monitoring tools to each cash register database or server in each store. While the method used in this research is data collection, among others in the form of observation and interviews as well as system development methods which include analysis, design, implementation, operation, and system maintenance. The results obtained from the design and connectedness of the system indicate that tax transaction monitoring can be carried out continuously so that the system can make it easier for taxpayers to carry out reporting.

Keywords: *online tax, taxpayers, transaction monitoring, auto reboot, internet of things*

ABSTRAK

Pembayaran pajak secara online merupakan salah satu cara untuk mempercepat proses pelaporan pajak bagi wajib pajak. Pengembangan sistem online masih sering mengalami masalah terutama bertumpuknya data yang akhirnya sistem online tersebut mengalami hambatan yang mengakibatkan proses pembayaran menjadi terhambat. Permasalahan lainnya yang mungkin terjadi masih banyak wajib pajak yang belum mengetahui dan memahami benar sistem online tersebut. Pengembangan transaction monitoring mempunyai kelemahan dimana koneksi sistem komputer akan mati jika digunakan secara terus menerus untuk memantau transaksi yang terjadi dalam kurun waktu 1 atau 2 minggu tergantung pada kondisi suhu sistem tersebut ditempatkan. Oleh karena itu, perlu adanya penambahan fitur auto reboot. Pengembangan transaction monitoring pajak berbasis internet of things (IoT) akan memudahkan wajib pajak membayar pajak karena adanya link antara wajib pajak dengan perusahaan yang bergerak dalam jasa pembayaran pajak sehingga wajib pajak tidak direpotkan lagi untuk urusan pembayaran pajak tersebut. Tujuan penelitian ini adalah merancang cara kerja transaction monitoring berbasis IoT dan menghubungkan alat transaction monitoring ke setiap database mesin kasir atau server di setiap toko. Sedangkan metode yang digunakan dalam penelitian ini adalah pengumpulan data antara lain berupa observasi dan wawancara serta metode pengembangan sistem yang meliputi analisis, design, implementasi, operasi, dan pemeliharaan sistem. Hasil yang diperoleh dari perancangan dan keterhubungan sistem menunjukkan bahwa transaction monitoring pajak dapat dijalankan secara kontinyu sehingga sistem tersebut dapat lebih mempermudah wajib pajak untuk melaksanakan pelaporan.

Kata kunci : *pajak online, wajib pajak, transaction monitoring, auto reboot, internet of things*

1. Pendahuluan

Undang-undang Nomor 28 Tahun 2009 tentang Pajak Daerah dan Retribusi Daerah menyatakan bahwa pajak daerah selanjutnya disebut pajak adalah kontribusi wajib kepada Daerah yang terutang oleh orang pribadi atau badan yang bersifat memaksa berdasarkan Undang-Undang, dengan tidak mendapatkan imbalan secara langsung dan digunakan untuk keperluan Daerah bagi sebesar-besarnya kemakmuran rakyat. Oleh karena itu, pajak memiliki kontribusi yang besar tidak hanya bagi Negara tapi juga bagi suatu daerah (Suardani, Purnamawati, dan Kurniawan., 2017). Pajak Daerah tersebut meliputi antara lain tempat makan, hotel, tempat hiburan dan tempat parkir. Pembayaran pajak daerah tersebut dilakukan oleh wajib pajak setiap bulannya berdasarkan rekapan manual yang sudah direkap oleh wajib pajak itu sendiri.

PT. Pinsi Elektra merupakan salah satu perusahaan yang bergerak di bidang IT dan Industri Telekomunikasi yang pada tahun 2019 telah mengembangkan sistem *online* pajak menggunakan *Transaction Monitoring* (TM) untuk pengawasan transaksi pajak bagi wajib pajak (WP). Perusahaan tersebut dapat memonitor data transaksi yang ada di komputer kasir WP. TM harus dipasang dan dihubungkan ke komputer kasir dan data transaksi tersebut langsung dikirim ke dinas pendapatan daerah agar bisa langsung di pantau secara *online* untuk mencegah WP (pelaku usaha) berbohong terkait omzet yang didapat.

Kemajuan di bidang teknologi, komputer, dan telekomunikasi mendukung perkembangan transaksi perdagangan dan kegiatan lainnya seperti pembayaran pajak. Pelaku bisnis tidak lagi mengalami kesulitan dalam memperoleh informasi apapun untuk menunjang aktivitas bisnisnya dengan menggunakan internet (Irmawati, 2011).

Kendala yang dialami dari sistem adalah *online* masih sering terjadi bertumpuknya data yang akhirnya sistem *online* tersebut mengalami hambatan yang mengakibatkan proses pembayaran menjadi terhambat. Permasalahan lainnya yang mungkin terjadi masih banyak wajib pajak yang belum mengetahui dan memahami benar sistem online tersebut.

Pengembangan *transaction monitoring* pajak berbasis *internet of things* (IoT) akan memudahkan wajib pajak membayar pajak karena adanya *link* antara wajib pajak dengan perusahaan yang bergerak dalam jasa pembayaran pajak sehingga wajib pajak tidak direpotkan lagi untuk urusan pembayaran pajak tersebut (Efendi, 2018).

Pengembangan *transaction monitoring* mempunyai kelemahan dimana koneksi sistem komputer akan mati jika digunakan secara terus menerus untuk memantau transaksi yang terjadi dalam kurun waktu 1 atau 2 minggu tergantung pada kondisi suhu sistem tersebut ditempatkan. Oleh karena itu, perlu adanya penambahan fitur *auto reboot*.

A. Perumusan Masalah

Berdasarkan latar belakang yang telah diuraikan diatas dapat dirumuskan masalah, yaitu :

- Bagaimana merancang *transaction monitoring* berbasis IoT?
- Bagaimana cara membuat alat *transaction monitoring* dapat terhubung ke mesin kasir atau server?
- Bagaimana melakukan pemasangan *Transaction Monitoring Device* (TMD) dengan *script auto reboot*?

B. Batasan Masalah

Batasan masalah dalam penelitian ini dapat dinyatakan sebagai berikut :

- Mengirimkan hasil rekaman data transaksi melalui GPRS menggunakan *Internet of Things* (IoT).
- Menggunakan *Linux Ubuntu* sebagai sistem operasi.
- Menggunakan *script auto reboot* untuk merestart TMD secara otomatis.

C. Tujuan dan Manfaat

Tujuan yang ingin dicapai dari penelitian ini adalah sebagai berikut :

- Merancang cara kerja *transaction monitoring* berbasis IoT.
- Menghubungkan alat *transaction monitoring* ke setiap *database* mesin kasir atau server di setiap toko.
- Merancang *script auto reboot* pada TMD.

Sedangkan manfaat penelitian ini adalah adanya alat *transaction monitoring* yang memberikan pengetahuan penting cara kerja untuk memonitor data transaksi dan dengan adanya *script auto reboot* diharapkan TMD akan merestart secara otomatis sesuai waktu yang sudah di tentukan.

D. Landasan Teori

Landasan teori akan berisikan materi-materi dan berkaitan dengan penelitian ini adalah sebagai berikut :

1. Monitoting Transaksi Pajak

Monitoring transaksi (*transaction monitoring*) pajak atau disebut Monitoring Pelaporan Pembayaran Pajak (MP3) merupakan sarana bagi Direktorat Jenderal Pajak (DJP) dalam mengawasi pembayaran pajak secara elektronik yang disebut e- payment. Sistem pembayaran pajak ini dilakukan oleh PT Pos (Persero), Bank Persepsi/ Bank Devisa Persepsi yang telah melakukan hubungan pertukaran informasi data secara online dengan Direktorat Jenderal Pajak, pembayaran setoran pajak melalui sistem pembayaran online dapat dilaksanakan melalui PT.Pos Indonesia (Persero) atau teller Bank Persepsi dan Bank Devisa Persepsi secara online maupun menggunakan fasilitas alat transaksi yang disediakan oleh bank persepsi dan Bank Devisa Persepsi *online*.

2. *Internet of things*

Internet of Things disingkat dengan IoT adalah sebuah sistem yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus yang dihubungkan mesin, peralatan, dan benda fisik lainnya dengan sensor jaringan dan aktuator untuk

memperoleh data dan mengelola kinerjanya sendiri yang memungkinkan mesin untuk berkolaborasi dan bertindak berdasarkan informasi baru yang diperoleh secara independen. Sistem ini dapat berkomunikasi satu dengan yang lain sebagai bagian dari satu kesatuan sistem terpadu menggunakan jaringan internet sebagai penghubung. Perangkat IoT terdiri dari sensor sebagai media pengumpul data, sambungan internet sebagai media komunikasi, dan server sebagai pengumpul informasi yang diterima sensor dan untuk analisis (Efendi, 2018; Artiyasa, et. al., 2020).

3. General Packet Radio Service (GPRS)

GPRS merupakan teknologi yang memungkinkan pengiriman dan penerimaan data lebih cepat dibandingkan dengan penggunaan teknologi *Circuit Switch Data* atau CSD. Penggabungan layanan telepon seluler dengan *General Packet Radio Service* (GPRS) menghasilkan generasi baru yang disebut 2.5G. Sistem GPRS dapat digunakan untuk transfer data (dalam bentuk paket data) yang berkaitan dengan e-mail, data gambar (MMS), *Wireless Application Protocol* (WAP), dan *World Wide Web* (WWW).

4. Wireless Fidelity (WIFI)

Wifi, yang merupakan singkatan dari *Wireless Fidelity* adalah sebuah teknologi penghantar komunikasi data tanpa kabel yang dapat digunakan untuk berkomunikasi dan mengirim data dengan kecepatan yang tinggi. Kecepatannya bisa mencapai 10, 100, sampai 1000 Mbps. Teknologi wifi menggunakan sinyal radio di frekuensi tertentu, sehingga wajar jika kecepatan transfer data sangat cepat.

5. Local Area Network (LAN)

LAN merupakan jaringan yang bisa mentransfer file yang besar dengan kecepatan tinggi. Kabel LAN biasanya digunakan oleh banyak komputer dalam waktu bersamaan, dengan kabel LAN ini bisa menggantikan flashdisk dengan tingkat kecepatan yang lebih tinggi.

6. SMS

SMS digunakan untuk memberikan konfirmasi ke Badan Pengelolaan Pajak Daerah (BPPD) untuk mengetahui total yang harus dibayarkan oleh wajib pajak (WP) di setiap bulannya.

7. Linux Ubuntu

GNU/Linux, termasuk Ubuntu, adalah sistem operasi berdasarkan lisensi sumber terbuka. Ini berarti bahwa sistem operasi dapat digunakan secara gratis untuk semua orang dan tidak perlu membeli lisensi seperti sistem operasi Windows. Selain itu, setiap orang bebas berkontribusi pada pengembangan sistem operasi ini untuk meningkatkan di masa depan. Maka dari itu sistem operasi dari alat transaksi monitoring menggunakan Linux Ubuntu.

8. Cron dalam Script auto reboot

Cron adalah *tool* di sistem operasi berbasis UNIX (**Linux, Ubuntu**, dan lain-lain) yang berfungsi untuk menjalankan *task* atau script secara otomatis. *Cron job* adalah sebutan penggunaan *cron* untuk menjadwalkan *task* di waktu yang sudah ditentukan secara berulang kali. Pada penggunaannya, *cron job* tidak hanya dapat dimanfaatkan untuk backup, pemeliharaan *server*, dan

keperluan teknis hosting saja. Bagi pemilik *blog* misalnya, fitur ini juga dapat digunakan untuk memeriksa *broken link* secara berkala atau mengirim pesan ke subscriber. *Cron* dapat digunakan dengan mengetikkan perintah tertentu. Beri kut ini adalah perintah-perintah dasar yang perlu Anda ketahui. Setiap *cron job* diwakili oleh sebuah file *crontab*. Oleh karena itu gunakan perintah *crontab* untuk membuat file *crontab* baru (Muarif and Irwan (2017)).

E. Metode Penelitian

Metode penelitian ini akan membicarakan tentang bahan dan peralatan, metode pengumpulan data, dan metode pengembangan sistem

1. Bahan dan Peralatan

Dalam melakukan pemasangan TMD dibutuhkan alat dan bahan berupa perangkat keras dan perangkat lunak. Alat dan bahan yang dibutuhkan adalah sebagai berikut :

- a. Software PuTTY
- b. Software FileZilla
- c. Software Winbox/VMS Dashboard
- d. Kabel LAN
- e. Usb Modem
- f. Switch/Hub
- g. Extension USB
- h. Adaptor 5V/2A

F. Metode Pengumpulan Data

Tahapan pengumpulan data pada penelitian ini adalah sebagai berikut :

1. Observasi

Observasi dilakukan pada penelitian ini dilakukan di PT. Pini Elekra dengan mengamati dan mempelajari kondisi dan infrastruktur yang berkaitan dengan bahan dan peralatan yang tersedia.

2. Wawancara

Melakukan wawancara kepada manager area PT. Pini Elekra untuk mendapatkan informasi tentang mengenai penerapan *Transaction Monitoring* berbasis IoT di PT. Pini Elekra. Wawancara dilakukan dengan tujuan untuk mencari data dan informasi, sehingga akan diperoleh data dan informasi yang dibutuhkan dalam pembuatan sistem.

G. Metode Pengembangan Sistem

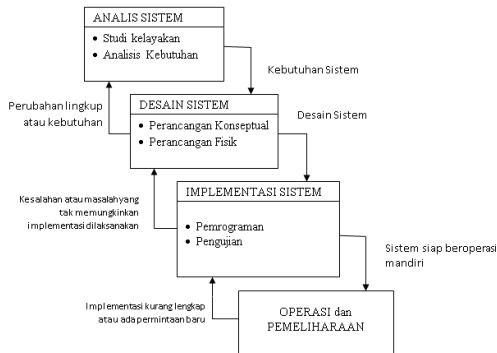
Pengembangan sistem yang digunakan adalah menggunakan metode SDLC (*System Development Life Cycle*). SDLC merupakan metode yang digunakan untuk merancang, membangun, memelihara, dan mengembangkan suatu sistem. Peneliti menggunakan metode SDLC ini karena bahwa ada beberapa tahapan-tahapan yang harus dilalui dalam merancang dan mengembangkan sebuah sistem. Tahapan tersebut sebagaimana diperlihatkan Gambar 1.

Tahapan yang dilakukan dalam pengembangan sistem sebagai berikut :

1. Analisis sistem

Merupakan tahapan untuk menganalisis kebutuhan apa saja yang ada di sistem toko/wajib pajak yang akan dipasang alat *transaction monitoring* dan meminta untuk

pengisian form parameter database aplikasi toko/wajib pajak. Tahap analisis sistem merupakan tahapan dimana sistem yang sedang berjalan dipelajari dan sistem baru diusulkan, sehingga isi dari tahap ini adalah menentukan hal detail terkait apa saja yang akan dikerjakan oleh sistem yang diusulkan. Analisis sistem mencakup kegiatan studi kelayakan dan analisis kebutuhan.



Gambar 1. Skema SDLC

2. Desain sistem

Merupakan tahapan untuk kebutuhan berupa perancangan konsep dimana alat *transaction monitoring* akan dipasang dan perancangan konsep fisik apa saja alat yang perlu di tambahkan untuk proses pemasangan alat *transaction monitoring* di toko/wajib pajak tersebut. Tahapan desain adalah tahapan mengubah kebutuhan yang masih berupa konsep menjadi spesifikasi sistem yang riil. Tahapan desain sistem dapat dibagi menjadi 2 tahap, yaitu perancangan konseptual dan perancangan fisik.

a) Perancangan Konseptual

Perancangan Konseptual seringkali disebut dengan perancangan logis. Pada kegiatan ini akan dijelaskan deskripsi secara detail dari spesifikasi sistem yang akan dibuat, yaitu meliputi *input* (data apa saja yang menjadi *input*), *output* (informasi apa saja yang menjadi *output*).

b) Perancangan Fisik

Pada perancangan fisik, rancangan yang bersifat konseptual diterjemahkan ke dalam bentuk fisik yang berupa laporan terkait beberapa hal berikut :

- 1) Rancangan Platform, berupa rancangan untuk menentukan software dan hardware yang digunakan.
- 2) Rencana Pengujian, berupa rencana yang dipakai dalam menguji sistem.

3. Implementasi sistem

Dalam tahapan ini peneliti akan melakukan integrasi alat *transaction monitoring* ke dalam database aplikasi toko/wajib pajak. Dalam tahap implementasi ada beberapa kegiatan yang harus dilakukan, yaitu:

a) Pemrograman dan pengujian

Setelah melakukan perancangan fisik, maka peneliti mulai melakukan perograman. Pemrograman diartikan sebagai aktifitas untuk membuat sebuah program aplikasi yang digunakan di komputer agar dapat bekerja sesuai dengan tujuan. Sedangkan pengujian diartikan sebagai langkah-langkah yang dilakukan untuk menguji

sistem berdasarkan tingkat kesalahannya dan serta kesesuaian menu-menu yang disediakan terhadap kebutuhan.

b) Instalasi

Instalasi dilakukan dengan cara menginstall perangkat keras dan perangkat lunak yang dibutuhkan oleh sistem.

4. Operasi dan Pemeliharaan

Setelah sistem berjalan sepenuhnya, sistem tersebut masuk ke dalam tahapan operasi. Tahapan operasi merupakan tahapan dimana sebuah sistem yang telah dibangun oleh peneliti mulai diterapkan. Selama sistem beroperasi, pemeliharaan sistem tetap diperlukan, dengan alasan bahwa dikhawatirkan dalam masa pengujian sistem masih menyisakan masalah, sehingga dalam jangka panjang sistem harus senantiasa disempurnakan. Dalam tahapan ini alat *transaction monitoring* sudah terhubung ke database aplikasi toko/wajib pajak. Selanjutnya tinggal melakukan maintenance berkala untuk memantau kondisi alat *transaction monitoring* berjalan dengan norma.

2. Pembahasan

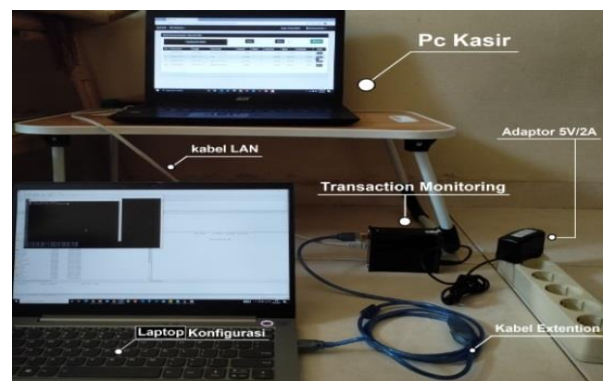
Hasil dan pembahasan dalam penelitian tersebut dilakukan untuk menjawab perumusan masalah dan tujuan penelitian sebagai berikut :

A. Merancang cara kerja *transaction monitoring* berbasis IoT

Perancangan cara kerja *transaction monitoring* berbasis IoT dapat dilakukan dengan tahapan sebagai berikut :

1. Implementasi *Transaction Monitoring*

Perekaman data pc kasir dapat terkoneksi maka *transaction monitoring* harus dikonfigurasi terlebih dahulu. Dengan dukungan software khusus, *transaction monitoring* dapat terhubung ke komputer kasir. Aplikasi Putty 0.73 dapat digunakan sebagai alat penghubung ke terminal linux di dalam *transaction monitoring* seperti terlihat pada Gambar 2.



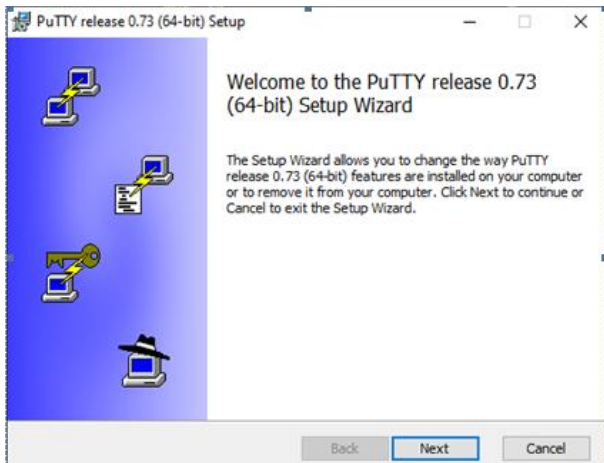
Gambar 2. Implementasi *Transaction Monitoring*

Komponen yang menyusun sistem tersebut adalah 1 unit laptop berfungsi sebagai pc kasir, 1 unit laptop berfungsi sebagai konfigurasi *transaction monitoring*, 1 buah kabel LAN sebagai media penghubung dari pc

kasir ke *transaction monitoring*, 1 buah adaptor 5V/2A untuk penghubung arus listrik ke *transaction monitoring*, 1 buah kabel *extension* sebagai penghubung *transaction monitoring* ke laptop konfigurasi.

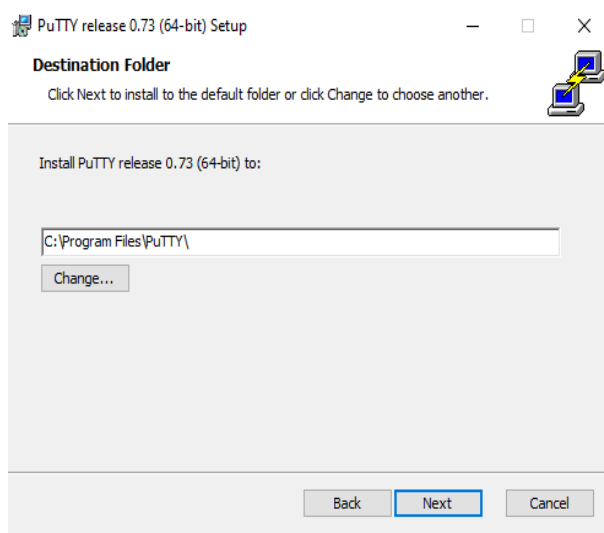
2. Instalasi Aplikasi Putty 0.73

Konfigurasi *transaction monitoring* dapat masuk ke terminal linux harus melakukan instalasi aplikasi putty 0.73 dengan melakukan protokol jaringan SSH, Telnet dan Rlogin, Klik Next untuk berpindah ke tahap instal selanjutnya. Tampilan menu pada instalasi aplikasi Putty 0,73 seperti terlihat pada Gambar 3.

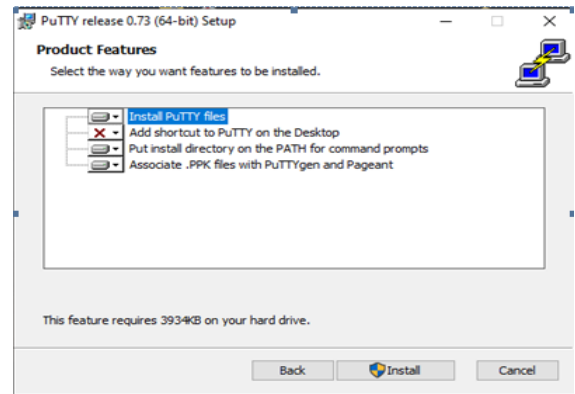


Gambar 3. Tampilan Menu pada Instalasi Aplikasi Putty 0.73

Selanjutnya melakukan letak penempatan folder dari aplikasi putty 0.73 yang akan di *install*. Pada kasus ini peletakan folder aplikasi putty 0.73 di Disk didalam folder **Program Files**.

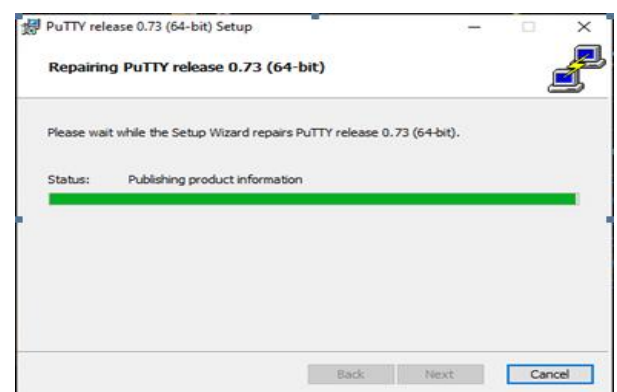


Gambar 4. Penempatan Folder Putty 0.73



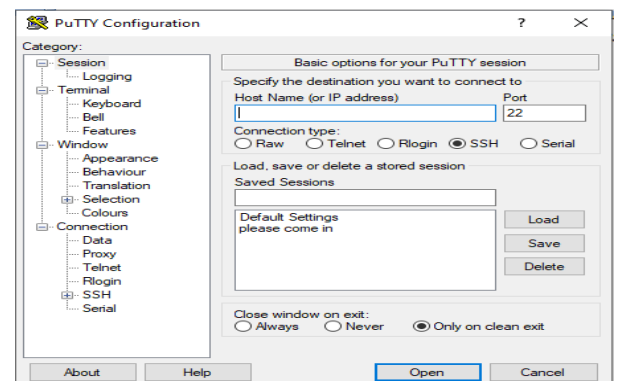
Gambar 5. Pemilihan Fitur Instalasi

Pada tahap ini pilih **Instal PuTTY files** lalu klik *Install*.



Gambar 6. Proses installasi aplikasi Putty 0.73

Setelah *installasi* selesai maka akan muncul tampilan awal aplikasi Putty. Selanjutnya dapat login menggunakan *Ip address/hostname* yang nantinya digunakan untuk bisa masuk ke dalam interface terminal *Transaction Monitoring*.



Gambar 7. Tampilan aplikasi Putty 0.73

Bentuk paling sederhana dalam sistem *transaction monitoring* adalah komputer kasir dan *transaction monitoring* terhubung melalui koneksi jaringan kabel LAN. Syarat dasar untuk mengadakan koneksi jaringan ke komputer kasirnya ialah dengan memasukan *ip address* yang ada di komputer kasirnya untuk di *input* didalam *transaction monitoring*, kedua perangkat komputer kasir dan *transaction monitoring* dapat melakukan uji koneksi dengan melakukan test *ping ip*

address ke ip masing-masing perangkat. Jika koneksi jaringan sudah terhubung selanjutnya melakukan penginputan akses parameter database aplikasi kasir ke dalam alat transaction monitoring. Syarat untuk penginputan parameter database aplikasi ke dalam transaction monitoring ialah user database, password database, port database dan table database. Jika semua syarat sudah lengkap selanjutnya tinggal melakukan integrasi database aplikasi.

Integrasi database merupakan penginputan akses database aplikasi ke dalam alat transaction monitoring, jika alat transaction monitoring dapat masuk ke dalam database aplikasi maka selanjutnya alat transaction monitoring sudah bisa memonitoring ke dalam table database aplikasi kasir dan sudah bisa melakukan penarikan data transaksi penjualan sesuai waktu yang dibutuhkan.

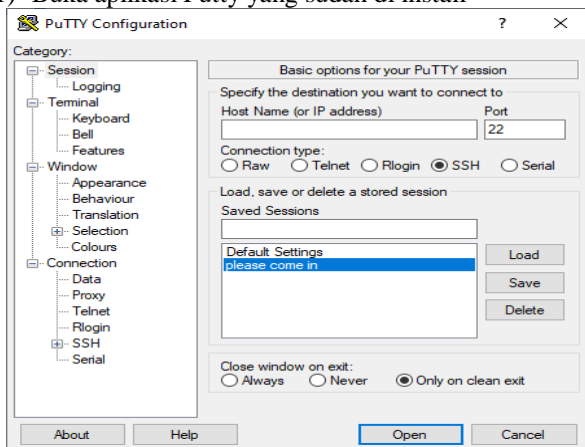
Transaction monitoring yang sudah terhubung ke database aplikasi dapat melakukan perekaman dan penarikan data transaksi perhari, perbulan dan pertahun. Hasil dari penarikan data transaksi dapat melihat diantaranya jumlah transaksi, nominal transaksi, dan tanggal transaksi secara real time.

B. Menghubungkan alat transaction monitoring ke setiap database mesin kasir atau server di setiap toko

Uji coba dilakukan menggunakan 1 komputer untuk konfigurasi transaction monitoring dan 1 komputer kasir yang sudah dibuatkan akses parameter database di bawah ini :

- a. User database : **coba**
- b. Password database : **coba123**
- c. Nama database : **tokoku_db**
- d. Kolom view : **tbl_transaksi**

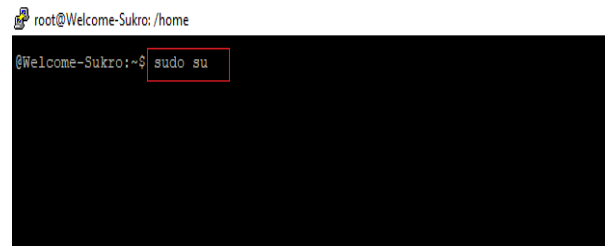
1) Buka aplikasi Putty yang sudah di install



Gambar 8. Masuk putty

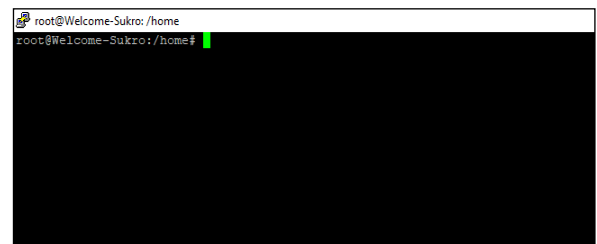
Jika aplikasi putty sudah terbuka silakan pilih tulisan *please come in* dan klik *Open* untuk masuk ke dalam terminal linux yang ada didalam transaction monitoring. 2) Jika sudah klik Open maka akan diarahkan ke tampilan awal terminal linux yang ada di dalam alat transaction monitoring, lalu akan beralih masuk ke dalam mode root/super user dengan cara memasukan sintaks ~\$ **sudo su** di dalam terminal linux dan klik

tombol *Enter*.



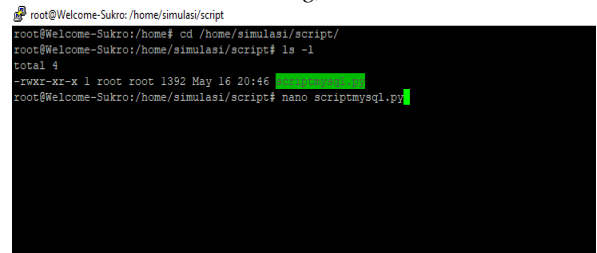
Gambar 9. Tampilan Terminal Linux

3) Setelah klik tombol *Enter*, terminal linux akan memunculkan direktori **/home#** dan saat ini user sedang berada di dalam direktori **/home#**.



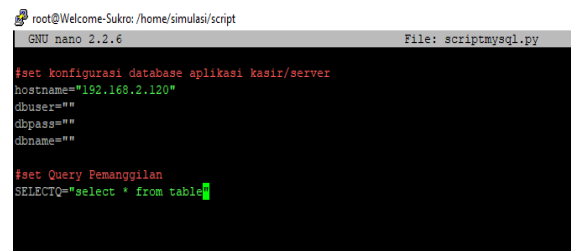
Gambar 10. Direktori Terminal

4) Selanjutnya masuk ke dalam script mysql yang ada di dalam transaction monitoring, dan tekan tombol **Enter**



Gambar 11. Pindah Direktori

5) Script mysql yang belum diisi selanjutnya kita akan mengisinya dengan parameter akses database aplikasi yang sudah dibuat seperti pada Gambar 12.



Gambar 12. Script mysql belum diisi

Gambar 13 adalah tampilan script mysql di terminal linux yang berada didalam alat transaction monitoring yang sudah di isi.

```

root@Welcome-Sukro:/home/simulasi/script
GNU nano 2.2.6 File: scriptmysql.py

#set konfigurasi database aplikasi kasir/serve:
hostname="192.168.2.120"
dbuser="coba"
dbpass="coba123"
dbname="tokoku_db"

#set Query Pemanggilan
SELECTQ="select * from tbl_transaksi"
    
```

Gambar 13. Script mysql sudah diisi

6) Lalu masukkan script mysql di dalam crontab untuk mempermudah pencarian dan pemanggilan script mysql.

```

root@Welcome-Sukro:/home/simulasi/script
root@Welcome-Sukro:/home/simulasi/script# crontab -l

##### Selamat datang Dev Sukro nando darfero #####

# m h dom mon dow   command

#Untuk melakukan penarikan data silakan klik script di bawah ini
15 * * * * /home/simulasi/script/scriptmysql.py

##### SELAMAT DATANG #####
root@Welcome-Sukro:/home/simulasi/script#
    
```

Gambar 14. Salin nama script di dalam crontab

7) Tampilan alat transaction monitoring selesai integrasi dan berhasil masuk ke dalam database aplikasi kasir seperti pada Gambar 15.

```

root@Welcome-Sukro:/home/simulasi/script# mysql -u root -p
mysql: [Warning] Using a password on the command line to connect is insecure. Use --password= or --prompt='Password: ' to avoid this warning. Use --help to learn more.
mysql> use tokoku_db;
mysql> select * from tbl_transaksi;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| no_transaksi | waktu      | id_kasir | subtotal | diskon | total_akhir | bayar | kembalian | no_transaksi | waktu      | id_kasir | subtotal | diskon | total_akhir | bayar | kembalian |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1502017001 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017001 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017002 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017002 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017003 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017003 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017004 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017004 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017005 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017005 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017006 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017006 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017007 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017007 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017008 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017008 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017009 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017009 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017010 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017010 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
mysql>
    
```

Gambar 15. Integrasi database selesai

Uji coba penarikan data hasil rekaman transaksi digunakan untuk melihat jumlah transaksi perhari sehingga bisa memantau hasil rekaman dari alat transaction monitoring yang telah terhubung ke aplikasi kasirnya. Adapun kolom database yang masuk dalam perekaman transaction monitoring ialah “no_transaksi, tgl_transaksi, waktu, id_kasir, subtotal, diskon, total_akhir, bayar dan kembalian”, seperti terlihat pada Gambar 16.

```

root@Welcome-Sukro:/home/simulasi/script# mysql -u root -p
mysql: [Warning] Using a password on the command line to connect is insecure. Use --password= or --prompt='Password: ' to avoid this warning. Use --help to learn more.
mysql> use tokoku_db;
mysql> select * from tbl_transaksi;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| no_transaksi | waktu      | id_kasir | subtotal | diskon | total_akhir | bayar | kembalian | no_transaksi | waktu      | id_kasir | subtotal | diskon | total_akhir | bayar | kembalian |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1502017001 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017001 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017002 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017002 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017003 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017003 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017004 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017004 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017005 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017005 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017006 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017006 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017007 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017007 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017008 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017008 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017009 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017009 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
| 1502017010 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 | 1502017010 | 2017-04-17 20:17:04 | 114197 | 40000 | 0 | 40000 | 40000 | 0 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
mysql>
    
```

Gambar 16. Perekaman Transaction Monitoring

C. Merancang script auto reboot pada TMD

Uji coba menggunakan script yang di tanam di terminal linux dengan membuat akses agar script auto reboot bisa dijalankan didalam transaction monitoring dan untuk menjalankan script auto reboot secara otomatis harus memindahkan nama folder auto reboot.sh kedalam crontab agar bisa diatur waktu berjalannya secara otomatis. Setingan waktu di crontab menggunakan lima bintang dan arti pada setiap bintangnya dimana bintang pertama adalah detik, bintang kedua adalah menit, bintang ketiga ialah jam, bintang keempat ialah hari dan bintang kelima adalah bulan. Setingan auto reboot pada penelitian ini menggunakan setiap 20 menit sekali transaction monitoring akan berjalan secara otomatis. Proses pembuatan script auto reboot ditunjukan pada Gambar 17- Gambar 21.

```

root@Welcome-Sukro:/home/restart
root@Welcome-Sukro:/home/restart# ls -l
total 24
drwxr-xr-x 2 root root 4096 May 16 20:46 monitor
drwxr-xr-x 2 root root 4096 May 16 12:51 restart
drwxr-xr-x 3 root root 4096 May 16 12:42 simulasi
root@Welcome-Sukro:/home# cd restart
root@Welcome-Sukro:/home/restart#
    
```

Gambar 17. Berpindah direktori

```

root@Welcome-Sukro:/home/restart
root@Welcome-Sukro:/home/restart# ls -l
total 4
-rwxr-xr-x 1 root root 23 May 16 12:51 autoreboot.sh
root@Welcome-Sukro:/home/restart#
root@Welcome-Sukro:/home/restart# nano autoreboot.sh
    
```

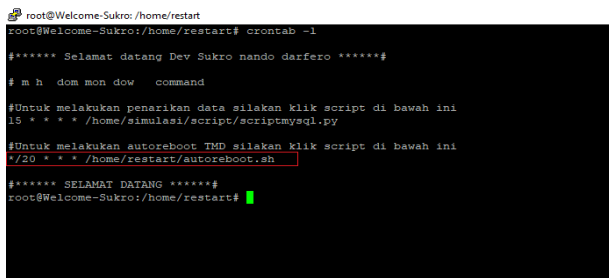
Gambar 18. membuat folder .sh

```

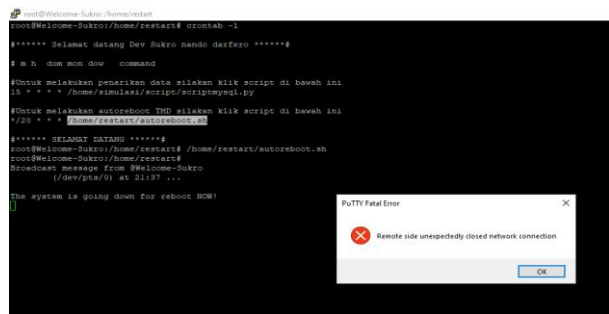
root@Welcome-Sukro:/home/restart
GNU nano 2.2.6 File: autoreboot.sh

#!/bin/sh
/sbin/reboot
    
```

Gambar 19. Menginput script auto reboot



Gambar 20. Menjalankan script auto reboot



Gambar 21. script auto reboot berhasil running

3. Penutup

A. Kesimpulan

Berdasarkan uraian sebelumnya, dapat diambil kesimpulan sebagai berikut :

1. Pada sistem *transaction monitoring* untuk dapat terkoneksi ke aplikasi kasir dibutuhkan akses parameter database agar *transaction monitoring* bisa terhubung ke aplikasi kasir
2. Dari hasil pengujian rekaman data transaksi bisa dipastikan hasil dari rekaman tersebut bisa dilihat secara langsung di dalam *transaction monitoring* langsung.
3. Dengan script auto reboot *transaction monitoring* akan melakukan restart secara otomatis sesuai waktu yang ditentukan, gunanya untuk mencegah alat *transaction monitoring* bekerja secara terus menerus.
4. Sistem *transaction monitoring* dapat diimplementasikan didalam jaringan *localhost*.

B. Saran

Beberapa saran untuk lebih optimalnya penelitian ini adalah sebagai berikut :

1. Akses database aplikasi kasir harus di buatkan khusus untuk *transaction monitoring* sehingga tidak mengganggu tabel database lainnya.
2. Sebaiknya ada dashboard khusus untuk memantau proses rekaman *transaction monitoring* sehingga jika ingin melihat rekaman transaksi tidak perlu untuk membuka alat *transaction monitoring* cukup dengan membuka dashboard saja.
3. Saran bagi peneliti berikutnya diharapkan supaya dapat mengembangkan *transaction monitoring* untuk jangkauan lebih luas sehingga tidak hanya sebatas pada koneksi lokal saja.

Daftar Pustaka

Undang-undang Nomor 28 Tahun 2009 tentang Pajak Daerah dan Retribusi Daerah

Artiyasa, M., Kusumah, I. H., Firmansyah F., Efendi, M. A., and Iriyanto, M., 2020. *Studi Perbandingan Platform Internet of Things (IoT) untuk Smart Home Kontrol Lampu Menggunakan NodeMCU dengan Aplikasi Web Thingspeak dan Blynk*. Jurnal Fidelity Vol. 02, No. 1, 2020, 03-10 59 dikirim 21 Februari 2020, direvisi 27 Februari 2020, diterima untuk publikasi 10 Maret 2020

Efendi, Y., 2018. *Internet of Things (IoT) Sistem Pengendalian Lampu Menggunakan Raspberry PI Berbasis Mobile*. Jurnal Ilmiah Ilmu Komputer, Vol. 4, No. 1, April 2018 (P) ISSN 2442-4512 (O) ISSN 2503-3832

Irmawati, D., 2011. *Pemanfaatan e-commerce dalam Dunia Bisnis*. Jurnal Ilmiah Orasi Bisnis-ISSN: 2085-1375 Edisi Ke-VI, November 2011.

Muarif, M. I., and Irwan, D., (2017). *Sistem Aouto Backup Elektronik Mail pada Mail Server Menggunakan Cron Job*. Jurnal Penelitian Ilmu Komputer, Sistem Embedded & Logic 5(2): 79-90 (2017) 79.

Suardani, N. L. P. R., Purnamawati, I. G. A., and Kurniawan, P. S., 2017. *Efektivitas Pelaksanaan Monitoring Pajak Daerah Berbasis Sistem Informasi pada Industri Perhotelan di Kabupaten Buleleng*, e-Journal S1 Ak Universitas Pendidikan Ganesha Jurusan Akuntansi Program S1 Vol: 8 No: 2 Tahun 2017.