



# Rekonsiliasi Data Keuangan Pemerintah *Pasca* Penerapan *Single Database Sakti* Menggunakan Finite State Automata

Anjahul Khuluq<sup>1</sup>, Windu Gata<sup>2</sup>, Jordy Lasmana Putra<sup>3</sup>, Hafifah Bella Novitasari<sup>4</sup>

<sup>1,2,3</sup>Universitas Nusa Mandiri

<sup>4</sup>STMIK Bani Saleh

<sup>1,2,3</sup>Jalan Kramat Raya No. 18, Senen, Jakarta Pusat

<sup>4</sup>Jl Mayor Hasibuan 68 Bekasi

Email: <sup>1</sup>14210199@nusamandiri.ac.id, <sup>2</sup>windu@nusamandiri.ac.id,

<sup>3</sup>jordy.jlp@nusamandiri.ac.id, <sup>4</sup>hafifahbela@stmik-banisaleh.ac.id

## Abstract

*Reconciliation of government financial data is a process of pairing data between work unit data and State General Treasurer data to be able to produce accountable and credible central government financial reports. The current reconciliation process is carried out by uploading data by the work unit to the e-rekon application, which can be replaced by an automatic process when implementing the single SAKTI database. The use of Finite State Automata in providing validation when the reconciliation process is carried out has proven to be able to help make the Government Financial Data Reconciliation Process effective and efficient*

**Keywords:** *Reconciliation, Finite State Automata, Single Database, SAKTI, Central Government Financial Report*

## Abstrak

*Rekonsiliasi data keuangan pemerintah merupakan suatu proses penyandingan data antara data Satuan Kerja Pemerintah dengan data Bendahara Umum Negara untuk dapat menghasilkan laporan keuangan pemerintah pusat yang akuntabel dan kredibel. Proses rekonsiliasi saat ini dilakukan melalui upload data oleh Satuan Kerja Pemerintah ke aplikasi e-rekon dapat digantikan dengan proses otomatis saat penerapan single database SAKTI. Penggunaan Finite State Automata dalam memberikan validasi saat proses rekonsiliasi dilakukan terbukti dapat membantu membuat Proses Rekonsiliasi Data Keuangan Pemerintah yang efektif dan efisien*

**Kata kunci:** *Rekonsiliasi, Finite State Automata, Single Database, SAKTI, Laporan Keuangan Pemerintah Pusat*

## 1. PENDAHULUAN

Rekonsiliasi adalah proses membandingkan data keuangan antara pencatatan pada Bendahara Umum Negara dengan pencatatan yang ada di kementerian/lembaga sehingga pencatatan kedua belah pihak dapat diyakini kebenarannya. Rekonsiliasi merupakan salah satu hal yang sangat penting dalam penyusunan laporan keuangan untuk menghasilkan laporan keuangan yang akuntabel dan kredibel karena perbedaan data transaksi keuangan dapat dikurang [1]. Penyusunan Laporan Keuangan Pemerintah Pusat harus didahului dengan rekonsiliasi data transaksi keuangan antara Bendahara Umum Negara, dalam hal ini KPPN, dengan Satuan Kerja [2]. Rekonsiliasi

memiliki peran dalam meminimalkan terjadinya perbedaan pencatatan transaksi keuangan yang berdampak pada tingkat kevalidan dan keakuratan data yang dalam penyajian laporan keuangan [3].

Proses Rekonsiliasi dilakukan dengan cara Satuan Kerja sebagai pemilik data Sistem Akuntansi Instansi (SAI) melakukan perekaman transaksi melalui Aplikasi RKA-KL untuk penganggaran, Aplikasi SAS untuk pelaksanaan anggaran, Aplikasi Persediaan untuk penatausahaan dan Akuntansi Persediaan dan Aplikasi SIMAK-BMN untuk penatausahaan dan akuntansi aset tetap untuk selanjutnya dibuatkan ADK kirim ke Aplikasi SAIBA, selanjutnya Aplikasi SAIBA menerima ADK dari aplikasi- aplikasi tersebut dan melakukan transaksi terkait jurnal penyesuaian lalu membuat ADK kirim ke Aplikasi e-rekon LK untuk selanjutnya ADK tersebut diupload ke Aplikasi e-rekon LK untuk dibandingkan dengan data Sistem Akuntansi Umum (SAU) yang berasal dari Aplikasi SPAN pada Kementerian Keuangan selaku Bendahara Umum Negara. SPAN adalah bagian dari sistem informasi pengelolaan keuangan negara yang terintegrasi sebagai implementasi dari modernisasi anggaran dan perbendaharaan Negara [4].

Kondisi aplikasi yang terpisah dan adanya proses input maupun output data aplikasi dari satu ke yang lainnya menggunakan cara kirim dan terima data kurang efisien karena berpotensi timbulnya duplikasi data karena atas data yang sama dapat terdapat pada beberapa database yang berbeda sehingga proses penyusunan laporan keuangan pemerintah pusat tidak dapat cepat dilakukan dan juga tidak akurat [5]. Kementerian keuangan, melalui Ditjen Perbendaharaan, melakukan perbaikan dalam pengelolaan keuangan negara terutama terkait laporan keuangan yang akurat dan akuntabel dengan melakukan perubahan dari sisi Teknologi Informasi berupa pengembangan sistem baru untuk pengelolaan keuangan negara dengan nama Sistem Aplikasi Keuangan Tingkat Instansi (SAKTI) yang digunakan dalam mendukung Sistem Perbendaharaan dan Penganggaran Negara pada Instansi Pemerintah yang berisi 9 modul yang terintegrasi dalam satu aplikasi [6]. SAKTI ditujukan untuk dapat menggantikan aplikasi yang digunakan saat ini pada level Kementerian / Lembaga sampai dengan tingkat satker dengan topologi jaringan infrastruktur menggunakan *single database* yang dapat diakses secara *real time* oleh seluruh *user* di seluruh Indonesia [7]. Transaksi- transaksi yang ada pada SAKTI dapat dilakukan monitoring melalui Aplikasi MonSakti [8]. Pemantauan dan monitoring data transaksi SAKTI pada Aplikasi MonSakti memiliki tujuan agar beban SAKTI tidak terlalu berat sehingga bisa lebih fokus untuk digunakan dalam transaksi.

Dengan adanya satu database maka seharusnya proses rekonsiliasi sudah tidak perlu dilakukan menggunakan ADK dan bahkan sebelum proses rekonsiliasi dilakukan Satuan Kerja sudah dapat melihat hasil rekonsiliasi

serta mengetahui daftar transaksi yang harus diperbaiki (*to do list*) untuk segera dilakukan perbaikan pada transaksi tersebut sebelum melakukan proses rekonsiliasi sehingga hal tersebut dapat membuat proses rekonsiliasi lebih efektif dan efisien. Proses rekonsiliasi dengan sistem yang baru yang dapat digabungkan dalam Aplikasi MonSakti yang sudah ada saat ini untuk menggantikan Aplikasi e-rekon LK sehingga selain mengurangi banyaknya aplikasi, diharapkan juga dapat memberikan nilai tambah dalam penyusunan laporan keuangan pemerintah pusat.

*Finite State Automata (FSA)* terbukti dapat membantu perancangan sistem informasi pada beberapa penelitian sebelumnya, di antaranya adalah *FSA* dapat digunakan sebagai logika dasar suatu proses validasi pada sistem informasi permohonan sehingga diharapkan dapat mengurangi penggunaan waktu, serta mengurangi kesalahan dengan adanya validasi oleh sistem [9]. Pada penelitian lain juga memberikan solusi dalam permasalahan kwitansi manual yang dapat diselesaikan dengan e-kwitansi yang menggunakan *FSA* sebagai logika dasar dalam merancang sistem informasi pengajuan e-kwitansi sehingga dapat mengurangi kesalahan perekaman dan memudahkan validasi data [10]. Sedangkan pada penelitian lainnya, *FSA* dengan jenis *NFA* dapat menggambarkan siklus perkuliahan sehingga dapat membantu mahasiswa lulus tepat waktu [11].

Dalam penelitian ini dibuat rancangan proses rekonsiliasi dengan mengolah data pada Sistem Akuntansi Instansi (SAI) menggunakan SAKTI yang dijadikan sebagai input saat melakukan proses rekonsiliasi. Data yang akan digunakan sebagai input adalah tutup periode, hasil rekonsiliasi, penjelasan perbedaan rekonsiliasi (bila terdapat perbedaan), daftar *to do list* yang belum selesai pada periode rekonsiliasi, dan penjelasan *to do list* belum diselesaikan (bila terdapat *to do list*). Agar data yang diproses akurat sesuai dengan syarat proses rekonsiliasi maka sistem akan melakukan pengecekan *input* dan melakukan validasi data sebelum proses rekonsiliasi disimpan ke dalam sistem menggunakan *Finite State Automata (FSA)* agar proses rekonsiliasi yang dilakukan sesuai dengan persyaratan proses yang diperlukan.

## 2. METODOLOGI PENELITIAN

Penelitian ini menggunakan tahapan yang dibagi ke dalam 3 (tiga) tahap yaitu: (1) Identifikasi Masalah, (2) Pencarian Solusi Permasalahan, dan (3) Perancangan *Finite State Automata*.

Pada Gambar 1 memberikan penjelasan terkait tahapan penelitian yang dilakukan pada penelitian ini adalah sebagai berikut:

### a) Tahap Identifikasi Masalah

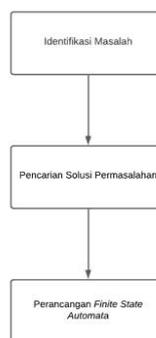
Pada tahap ini penulis melakukan identifikasi atas permasalahan terkait Proses Rekonsiliasi data Keuangan Pemerintah saat ini untuk dapat dicarikan solusi untuk menyelesaikan permasalahan tersebut.

b) Tahap Pencarian Solusi Permasalahan

Pada tahap ini penulis mencari solusi atas permasalahan tersebut dengan membuat alur proses rekonsiliasi sebagai perbaikan proses sebelumnya untuk selanjutnya dapat dilakukan perancangan menggunakan *Finite State Automata*.

c) Tahap Perancangan *Finite State Automata*

Pada tahap ini perancangan *Finite State Automata* dilakukan sesuai dengan alur proses rekonsiliasi yang sudah dibuat sebelumnya untuk memastikan bahwa permasalahan yang ada dapat diselesaikan dengan logika dasar komputasi.



**Gambar 1.** Tahap Penelitian

### 3. HASIL DAN PEMBAHASAN

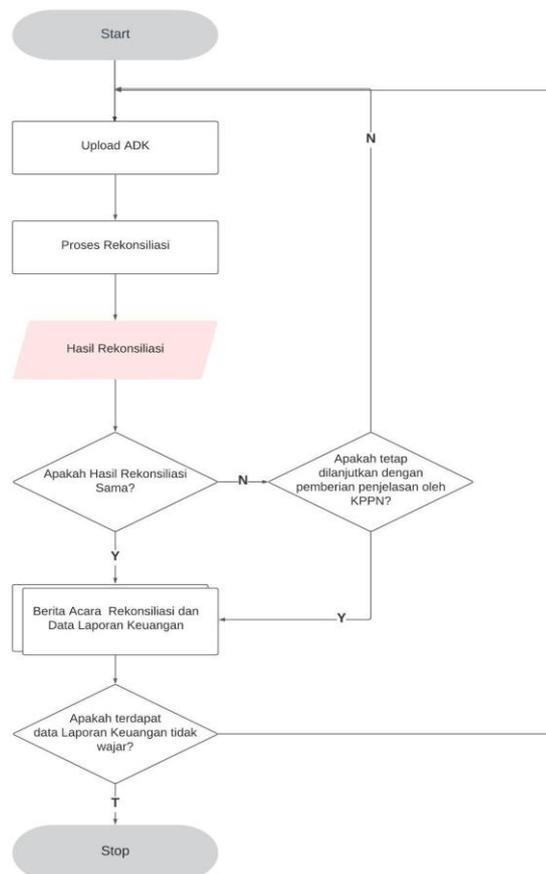
#### 3.1. Identifikasi Permasalahan

Permasalahan terkait rekonsiliasi adalah proses rekonsiliasi yang kurang efektif dan efisien karena hasil rekonsiliasi baru dihasilkan setelah proses rekonsiliasi dilakukan melalui upload ADK sehingga apabila terdapat perbedaan data atau saldo tidak wajar harus dilakukan proses rekonsiliasi berkali-kali sebagaimana gambar di bawah.

Proses Rekonsiliasi saat ini sebagaimana pada *flowchart* gambar 2 dilakukan setelah Satker melakukan upload data ke Aplikasi e-rekon LK di mana sistem akan melakukan proses rekonsiliasi dengan menyandingkan data Satker (SAI) dengan data SAU milik Bendahara Umum Negara untuk mendapatkan hasil rekonsiliasi. Dari hasil rekonsiliasi tersebut apabila terdapat perbedaan yang tidak dapat diterima KPPN maka Satker akan melakukan perbaikan transaksi lalu melakukan proses rekonsiliasi dari awal mulai dari upload ADK sampai mendapatkan hasil rekon yang sama atau hasil rekon yang dapat diterima. Meskipun hasil rekonsiliasi sudah sama tetapi apabila dari data Laporan Keuangan yang disajikan dalam Aplikasi e-rekon LK ditemukan ketidakwajaran saldo laporan maka Satker harus melakukan perbaikan transaksi lagi dan melanjutkan dengan proses rekonsiliasi dari awal lagi. Proses rekonsiliasi seperti ini tentu sangat tidak efektif dan efisien sehingga perlu ditemukan solusi untuk mendapatkan proses rekonsiliasi yang lebih baik.

Proses Rekonsiliasi yang dilakukan melalui *Upload ADK* memang tidak dapat dihindari dan sudah menjadi praktek terbaik dengan kondisi saat ini karena data SAI berada pada masing- masing satker, sedangkan data SAU berada pada Kementerian Keuangan sehingga Aplikasi e-rekon LK menjembatani penyandingan data SAI melalui *upload ADK* dengan data SAU yang diambil secara langsung dari database Kementerian Keuangan ke Aplikasi e-rekon LK.

Dengan adanya SAKTI yang memiliki konsep *single database* yang dikelola oleh kementerian keuangan, di mana artinya database pada SAI dan database pada SAU sama-sama berada pada Kementerian Keuangan maka permasalahan atas proses rekonsiliasi harusnya dapat diselesaikan dengan konsep yang baru dalam melakukan proses rekonsiliasi.

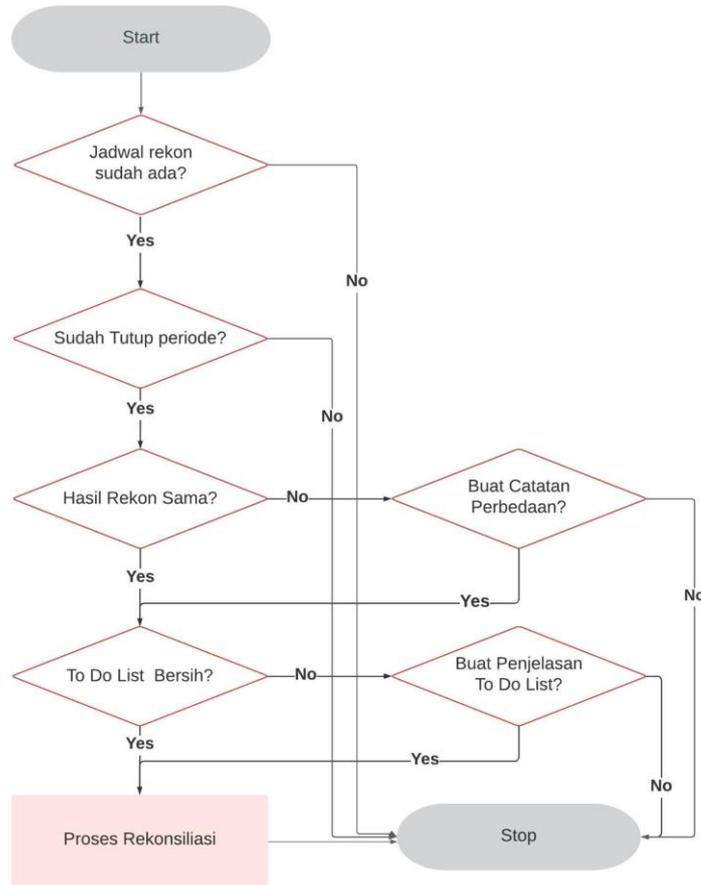


**Gambar 2.** Proses Rekonsiliasi saat ini

### 3.2. Solusi atas Permasalahan

Konsep yang dibangun dalam penelitian ini memanfaatkan *single database* pada Aplikasi SAKTI sebagai data SAI yang berada pada kementerian keuangan, dan data SAU pada Aplikasi SPAN yang juga berada pada Kementerian Keuangan sehingga Aplikasi yang digunakan dalam proses

rekonsiliasi dapat secara langsung mengambil data- data tersebut untuk disandingkan lebih awal sebagaimana dalam gambar 3.



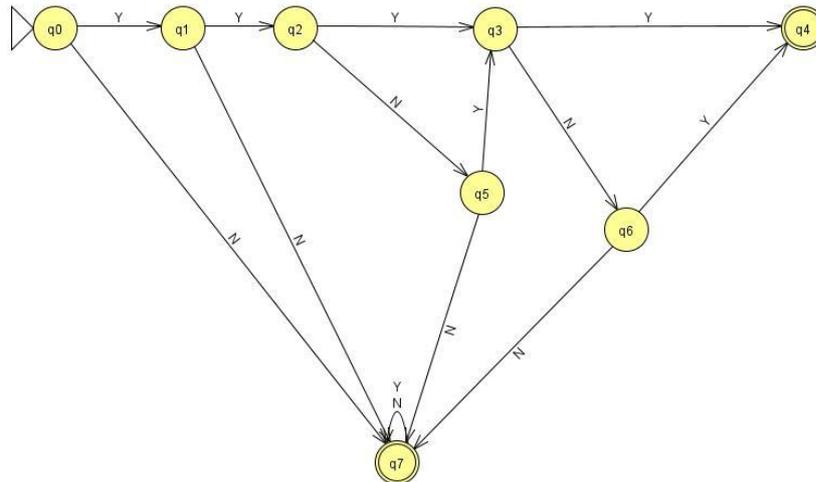
**Gambar 3.** Proses Rekonsiliasi yang diusulkan

Proses Rekonsiliasi yang diusulkan dalam penelitian ini sebagaimana dalam gambar 3 dilakukan tanpa upload data yaitu menggunakan Aplikasi MonSakti yang sudah terintegrasi dengan SAKTI (memuat data SAI) di mana saat Proses Rekonsiliasi akan dilakukan sudah terdapat pengecekan-pengecekan terhadap jadwal rekonsiliasi dan status tutup periode dan juga bahkan disediakan informasi hasil rekonsiliasi yang sudah dihasilkan dari penyandingan data SAI dan data SAU (hasil rekon sama atau beda) dan informasi *to do list* dari laporan keuangan berdasarkan data SAKTI yang sudah dibuatkan validasi data tidak wajar yang perlu diselesaikan (sudah diselesaikan atau belum) sehingga perbaikan data dapat dilakukan sebelum proses rekonsiliasi atau apabila tetap melanjutkan proses rekonsiliasi dengan data beda dan/atau *to do list* belum diselesaikan maka harus memberikan penjelasan perbedaan data ataupun *pending* penyelesaian *to do*

list tersebut sehingga proses rekonsiliasi dapat dilakukan dengan lebih cepat dan akurat.

### 3.2. Perancangan *Finite State Automata*

Perancangan *FSA* menggunakan *NFA* yang dibangun sebagaimana dalam gambar sebagai berikut:



**Gambar 4.** Diagram State NFA Proses Rekonsiliasi

Pada Gambar 4 menunjukkan rancangan diagram state yang menjelaskan proses rekonsiliasi data keuangan pemerintah. Secara formal *FSA* pada Gambar 2 dinyatakan dalam 5 (lima) tuple sebagai berikut:

- $\Sigma = \{Y,N\}$
- $Q = \{Q0,Q1,Q2,Q3,Q4,Q5,Q6,Q7\}$
- $S = \{Q0\}$
- $F = \{Q4\}$

**Tabel 1.** Himpunan State

State	Deskripsi
Q0	Jadwal Rekonsiliasi adalah <i>start state</i> yang merupakan titik awal dari pengajuan Proses Rekonsiliasi adalah pada saat jadwal rekonsiliasi dibuka
Q1	Status Tutup Periode apakah sudah dilakukan
Q2	Hasil rekon apakah sudah sama
Q3	Data to do list apakah sudah diselesaikan
Q4	Proses berhasil, adalah <i>final state</i> di mana apabila seluruh persyaratan dipenuhi maka Proses rekonsiliasi berhasil

State	Deskripsi
Q5	Penjelasan hasil rekon beda apakah ada
Q6	Penjelasan to do list belum selesai apakah ada
Q7	Proses ditolak

Tabel 1 menjelaskan tentang himpunan *state* yang ada pada perancangan diagram FSA.

Tabel 2. Himpunan Abjad

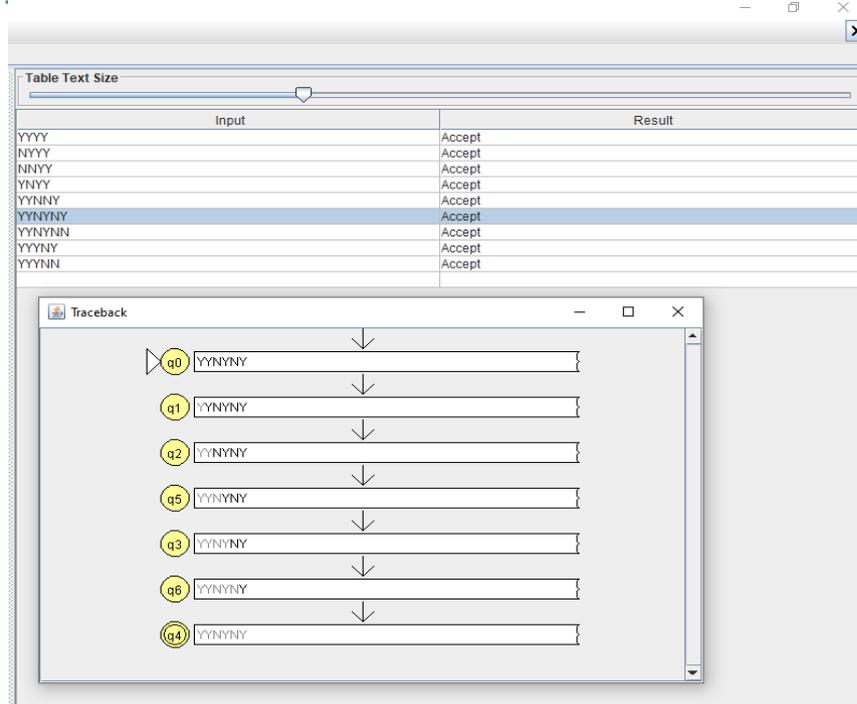
Abjad	Deskripsi
Y	Y adalah <i>Yes</i> untuk ke <i>state</i> berikutnya
N	N adalah <i>No</i> untuk ke <i>state</i> berikutnya

Tabel 2 menjelaskan tentang himpunan abjad yang ada pada perancangan diagram FSA.

Tabel 3. Relasi transisi

Transisi	Y	T
Q0	Q1	∅
Q1	Q2	Q7
Q2	Q3	Q5
Q3	Q4	Q6
Q4	∅	∅
Q5	Q3	Q7
Q6	Q4	Q7
Q7	∅	∅

Tabel 3 menjelaskan tentang perpindahan *state* yang ada pada perancangan diagram FSA apabila ada *input* yang masuk. sebagai contoh *state* Q0 mendapatkan *input* yang sesuai yaitu jadwal rekonsiliasi sudah ada maka akan masuk ke *state* Q1, namun apabila mendapatkan *input* yang tidak sesuai berupa informasi jadwal rekonsiliasi belum dibuka maka akan masuk ke *state* Q7. Perpindahan antar *input* bernilai Y apabila *input* sudah sesuai dan bernilai N apabila *input* tidak sesuai.



Gambar 5. Pengujian FSA

Pengujian input *string* dilakukan dengan menggunakan aplikasi JFLAP berdasarkan berbagai kondisi data yang sudah diubah menjadi *string* sebagaimana dalam gambar 5 terhadap seluruh pengujian berhasil diterima. Proses pengujian pada gambar 5 tersebut dapat dilihat dari contoh data yang berisi: Jadwal rekon sudah dibuka, Sudah tutup buku, hasil rekon beda, terdapat penjelasan perbedaan, to do list belum diselesaikan, terdapat penjelasan atas belum selesainya to do list sehingga dari data tersebut seharusnya proses rekonsiliasi berhasil. Dari data tersebut ketika dikonversi menjadi *string* memberikan *input* YYNYNY dan hasil pengujian menggunakan software *jflap* berhasil dan berhenti pada Q4 yang berarti proses berhasil.

#### 4. SIMPULAN

*Finite State Automata* dapat membantu menciptakan proses Rekonsiliasi data keuangan pemerintah *pasca* penerapan *single* database SAKTI dengan memberikan validasi terhadap data- data yang sudah tersedia yaitu tutup periode, hasil rekon, dan *to do list* saat proses rekonsiliasi dilakukan sehingga proses rekonsiliasi dapat berjalan dengan efektif dan efisien.

#### DAFTAR PUSTAKA

- [1] E. S. Purnomo and F. Samopa, "Pembuatan Sistem Informasi Rekonsiliasi Keuangan Negara Menggunakan PHP dan MySQL," *J. Tek. POMITS*, vol. 2, no. 2, 2013.

- [2] D. E. Limbong, "Rekonsiliasi Data Keuangan dan Karakterisasi Satuan Kerja dengan Teknik Data Mining," *Indones. Treas. Rev. J. Perbendaharaan Keuang. Negara dan Kebijak. Publik*, vol. 1, no. 3, 2016, doi: 10.33105/itrev.v1i3.48.
- [3] S. A. Syahdan, "Analisis Proses Rekonsiliasi Pada Kantor Pelayanan Perbendaharaan Negara (KPPN) Banjarmasin," *J. Manaj. dan Akunt.*, vol. 13, no. 1, 2012.
- [4] D. A. Wibawa and . S., "Analisis User Acceptance dan Usage Atas Implementasi SPAN pada Kementerian Keuangan," *Indones. Treas. Rev. J. Perbendaharaan Keuang. Negara dan Kebijak. Publik*, vol. 3, no. 4, 2018, doi: 10.33105/itrev.v3i4.78.
- [5] F. S. Samopa Ahmad, "Pengembangan Aplikasi Sistem Informasi Akuntansi dan Pengelolaan Keuangan Negara (SIAPKAN) Modul Aset Tetap Menggunakan Bahasa Pemrograman Java dan Database PostgreSQL," *J. Tek. ITS*, no. Vol 1, No 1 (2012): Jurnal Teknik ITS (ISSN 2301-9271), 2012.
- [6] H. Aditya, M. Ardiansyah, and W. Gata, "Pengelolaan Persediaan Pada Aplikasi Sakti Menggunakan Algoritma First In First Out (FIFO)," *J. Inform.*, vol. 20, no. 2, 2020.
- [7] S. Sudarto, "Pengembangan integrated financial management information system (IFMIS) di Indonesia," *Indones. Treas. Rev. J. Perbendaharaan Keuang. Negara dan Kebijak. Publik*, vol. 4, no. 2, 2019, doi: 10.33105/itrev.v4i2.127.
- [8] A. S. Marpaung, "Analisis Pemanfaatan Aplikasi 'Monitoring Sistem Aplikasi Keuangan Tingkat Instansi (Monsakti)' Dengan Metode Unified Theory Of Acceptance And Use Of Technology 2(UTAUT 2)," *Anal. Pemanfaat. Apl. "MONITORING Sist. Apl. Keuang. TINGKAT INSTANSI (MONSAKTI)" DENGAN Metod. UNIFIED THEORY Accept. USE Technol. 2(UTAUT 2)*, vol. 53, no. 9, 2013.
- [9] F. Said, D. Andriyanto, R. Sari, and W. Gata, "Perancangan Validasi Permohonan Narasumber Pada Sistem Informasi Permohonan Narasumber Menggunakan Finite State Automata," *Paradig. - J. Komput. dan Inform.*, vol. 22, no. 2, 2020, doi: 10.31294/p.v22i2.8157.
- [10] D. Irawan, R. Anugrah, M. Pratama, B. S. Prakoso, S. Rahayu, and W. Gata, "Perancangan Validasi Pengajuan E-Kwitansi Ziswaf State Automata," vol. 7, no. 2, pp. 39-44, 2021.
- [11] A. P. Giovani, F. Zamachsari, E. D. Agustono, M. I. Prasetya, and W. Gata, "Implementasi Finite State Automata Dalam Siklus Pembelajaran Magister Ilmu Komputer STMIK Nusa Mandiri," *CESS (Journal Comput. Eng. Syst. Sci.)*, vol. 5, no. 2, 2020, doi: 10.24114/cess.v5i2.16696.