

BAB I PENDAHULUAN

A. Latar Belakang

Transformasi digital dalam komputasi awan (*cloud computing*) menuntut efisiensi operasional yang tinggi, salah satunya melalui digitalisasi sistem manajemen tugas berbasis *Software as a Service* (SaaS). Model SaaS memungkinkan organisasi untuk menskalakan alur kerja secara dinamis tanpa terbebani biaya investasi perangkat keras awal yang besar [1]. Selain itu, model ini juga mengurangi kompleksitas instalasi dan pemeliharaan sistem karena seluruh infrastruktur dikelola oleh penyedia layanan [2]. Dengan demikian, SaaS menjadi salah satu pendekatan utama dalam meningkatkan efisiensi operasional dan fleksibilitas sistem informasi organisasi [3].

Untuk mendukung efisiensi tersebut, pengembangan sistem SaaS modern umumnya menerapkan arsitektur *multi-tenant*, yaitu suatu pendekatan yang memungkinkan banyak pengguna atau organisasi berbagi satu sistem yang sama dengan tetap mempertahankan isolasi data masing-masing [4]. Melalui implementasi komponen teknis pada lapisan data dan komputasi yang tepat, arsitektur ini mampu meningkatkan skalabilitas sistem hingga 70% serta menekan biaya operasional sebesar 60% [4]. Namun, penggunaan infrastruktur bersama (*shared infrastructure*) menjadikan mekanisme pengendalian akses sebagai salah satu aspek yang paling kritis dalam menjaga keamanan sistem. Setiap permintaan akses harus dipastikan hanya dapat menjangkau sumber daya yang menjadi hak pengguna sesuai dengan batas organisasi (*tenant boundary*) agar tidak terjadi kebocoran data antar-tenant (*cross-tenant data leakage*).

Permasalahan tersebut menjadi semakin kompleks pada sistem manajemen tugas yang melibatkan banyak pengguna dengan tanggung jawab berbeda dalam satu lingkungan *multi-tenant*. Pada sistem seperti ini, keputusan otorisasi tidak cukup hanya ditentukan berdasarkan peran (*role*) pengguna, tetapi juga dipengaruhi oleh hubungan pengguna terhadap tugas yang diakses (*ownership*), kondisi atau tahapan pekerjaan (*resource state*), serta batas organisasi tempat data berada (*tenant boundary*). Oleh karena itu, mekanisme kontrol akses yang mampu mempertimbangkan konteks operasional secara dinamis menjadi kebutuhan penting dalam sistem manajemen tugas berbasis *multi-tenant*.

Salah satu tantangan dalam arsitektur *multi-tenant* adalah menjaga isolasi data serta memastikan setiap akses dilakukan sesuai hak akses pengguna. Dalam praktiknya, model ini rentan terhadap kebocoran data lintas tenant dan pelanggaran otorisasi akibat miskonfigurasi hak akses. Permasalahan tersebut menunjukkan bahwa mekanisme kontrol akses tidak hanya memverifikasi identitas pengguna, tetapi juga mampu memvalidasi konteks akses terhadap sumber daya yang digunakan [5].

Sebagian besar sistem informasi konvensional masih menggunakan pendekatan *Role-Based Access Control* (RBAC) statis sebagai mekanisme pengelolaan hak akses pengguna. Meskipun RBAC efektif pada sistem yang sederhana, model ini memiliki keterbatasan dalam lingkungan *multi-tenant* karena keputusan otorisasi hanya didasarkan pada peran pengguna tanpa mempertimbangkan konteks sumber daya yang diakses [6]. Selain itu, peningkatan jumlah pengguna dan variasi kebutuhan akses dapat menyebabkan kompleksitas jumlah peran (*role explosion*) yang mempersulit tata kelola keamanan [7]. akibatnya, RBAC statis belum mampu mengakomodasi kebutuhan validasi berdasarkan kepemilikan tugas (*ownership*), status sumber daya (*resource state*), dan batas penyewa (*tenant boundary*) yang diperlukan dalam proses manajemen tugas *multi tenant*.

Seiring dengan berkembangnya kebutuhan keamanan sistem berbasis cloud, pendekatan keamanan modern mulai mengarah pada konsep *Zero Trust Architecture*. Pendekatan ini mengadopsi prinsip *never trust, always verify*, yaitu setiap permintaan akses harus divalidasi secara berkelanjutan berdasarkan identitas dan konteks akses [8]. Dengan demikian, paradigma *zero trust* menjadi landasan dalam pengembangan mekanisme otorisasi dinamis sehingga keputusan akses tidak hanya ditentukan oleh peran pengguna, tetapi juga mempertimbangkan *ownership*, *resource state*, dan *tenant boundary* pada setiap permintaan akses.

Kebutuhan akan isolasi data dan manajemen tugas yang terstruktur ditemukan pada PT Bharata Internasional Pharmaceutical. Sebagai perusahaan industri kosmetik dan obat herbal berskala nasional, perusahaan ini memiliki struktur organisasi kompleks yang terdiri dari 10 divisi operasional. Kompleksitas ini menuntut pemisahan hak akses yang sangat ketat. Dalam skema ini, setiap divisi mempresentasikan sebuah entitas penyewa (*tenant*) yang memerlukan isolasi data mandiri dalam satu ekosistem sistem manajemen tugas.

Berdasarkan hasil observasi, pengelolaan alur pengajuan tugas masih dilakukan secara manual melalui komunikasi lisan, aplikasi pesan instan seperti WhatsApp, dan pencatatan sederhana. Kondisi ini menyebabkan kesulitan dalam pelacakan tugas, koordinasi antar divisi, serta pengendalian aktifitas pengguna. Penelitian menunjukkan bahwa penggunaan media komunikasi informal dalam manajemen tugas mengakibatkan 70% proyek gagal mencapai akibat inefisiensi koordinasi dan kendala dalam pelacakan progres [9]. Selain berdampak pada efektivitas koordinasi, kondisi tersebut juga menyulitkan organisasi dalam menerapkan mekanisme otorisasi yang konsisten sehingga meningkatkan risiko akses tidak sah dan pelanggaran *tenant boundary* antar-divisi.

Meskipun berbagai platform manajemen tugas berbasis SaaS seperti Trello, Notion dan Asana telah menyediakan fitur kolaborasi dan pengelolaan proyek [10] [11], platform tersebut dikembangkan untuk memenuhi kebutuhan organisasi secara umum. Sementara itu, PT Bharata Internasional Pharmaceutical memerlukan

mekanisme otorisasi yang dapat disesuaikan dengan kebijakan internal perusahaan, khususnya dalam memvalidasi hak akses berdasarkan *role*, *ownership*, *resource state*, dan *tenant boundary*. Selain itu, penggunaan platform pihak ketiga juga membatasi fleksibilitas organisasi dalam mengembangkan mekanisme keamanan sesuai kebutuhan operasional perusahaan, serta menimbulkan ketergantungan terhadap penyedia layanan (*vendor lock-in*) dan keterbatasan kendali terhadap pengolahan data organisasi [12].

Permasalahan ini menjadi semakin relevan mengingat PT Bharata Internasional Pharmaceutical telah memiliki infrastruktur server internal yang mandiri. Ketersediaan tersebut seharusnya dapat dimanfaatkan untuk mengimplementasikan sistem kustom dengan model *deployment on-premise*. Langkah ini diambil untuk menjaga integritas dan keamanan data secara optimal dalam kendali penuh organisasi sekaligus memitigasi risiko kedaulatan data yang sering menjadi kendala pada platform SaaS public. Dengan demikian, diperlukan pengembangan sistem internal yang mengadopsi fleksibilitas fitur manajemen proyek sebagaimana dikembangkan dalam konsep SaaS [13], serta menerapkan mekanisme otorisasi yang mampu menyesuaikan keputusan akses berdasarkan konteks operasional dinamis.

Dalam pengembangan sistem manajemen tugas berbasis *multi-tenant*, mekanisme otorisasi merupakan aspek krusial karena miskonfigurasi pada bagian ini berpotensi memicu kebocoran data antar-divisi (*cross-tenant data leakage*). Berdasarkan studi literatur terhadap penelitian dalam lima tahun terakhir, ditemukan beberapa celah penelitian (*research gap*) yang menjadi landasan posisi penelitian ini.

Pertama, penelitian oleh [8] menekankan pentingnya paradigma *zero trust* dalam lingkungan *multi-tenant* untuk validasi berkelanjutan, namun focus penelitian tersebut masih berada pada kerangka kerja keamanan global di level infrastruktur jaringan, bukan pada implementasi logika otorisasi di tingkat operasional fungsional aplikasi. Kedua, penelitian oleh [9] telah merancang sistem manajemen proyek berbasis web dengan *multi-level* RBAC, tetapi mekanisme otorisasi yang digunakan masih bersifat statis karena hanya mempertimbangkan peran pengguna. Dengan demikian, masih diperlukan mekanisme otorisasi yang mampu mengintegrasikan validasi *role*, *ownership*, *resource state*, dan *tenant boundary* dalam proses pengambilan keputusan akses pada sistem manajemen tugas berbasis *multi-tenant*.

Berdasarkan permasalahan dan *research gap* tersebut, penelitian ini mengusulkan mekanisme *Dynamic Context-Aware Role-Based Access Control* (RBAC) sebagai model otorisasi dinamis yang mengintegrasikan validasi *role*, *ownership*, *resource state*, dan *tenant boundary* dalam proses pengambilan keputusan akses. Mekanisme tersebut diimplementasikan pada aplikasi manajemen

tugas berbasis web sebagai media implementasi untuk mengevaluasi model otorisasi yang diusulkan. Dengan demikian, kontribusi utama penelitian ini terletak pada pengembangan model otorisasi dinamis yang mampu menyesuaikan keputusan akses berdasarkan konteks operasional pada lingkungan *multi-tenant*.

Berdasarkan urgensi tersebut, penelitian ini diangkat dengan judul “Aplikasi Manajemen Tugas *Multi-Tenant* Berbasis Web Menggunakan Sistem *Dynamic Context Aware Role-Based Access Control*”.

B. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dalam penelitian ini adalah:

- a. Bagaimana membangun sistem manajemen tugas berbasis *multi tenant* sebagai solusi terhadap proses pengelolaan tugas konvensional di PT. Bharata Internasional Pharmaceutical?
- b. Bagaimana hasil pengujian mekanisme otorisasi *Dynamic Context-Aware Role-Based Access Control (RBAC)* berbasis tiga variabel konteks (*ownership, resource state, dan tenant boundary*) dalam mengelola otorisasi dan isolasi data antar divisi pada sistem manajemen tugas *multi tenant* PT Bharata Internasional Pharmaceutical?

C. Batasan Masalah

Batasan masalah membahas mengenai :

- a. Penelitian difokuskan pada pengembangan *backend* API menggunakan pendekatan *clean architecture*. Antarmuka pengguna (*frontend*) hanya digunakan sebagai media interaksi dan pengujian sistem jika dibutuhkan dalam pemaparan serta tidak menjadi fokus pengembangan maupun evaluasi penelitian.
- b. Implementasi arsitektur *multi-tenant* dibatasi pada model *Single Database – Shared Schema* pada basis data MongoDB untuk menjaga isolasi data antar divisi.
- c. Mekanisme otorisasi sistem dibatasi pada penerapan *Dynamic Context-Aware RBAC*, dengan evaluasi berbasis tiga variabel konteks : *Ownership, Resource State, dan Tenant Boundary*.
- d. Sistem autentikasi pengguna dikembangkan secara mandiri (*stateless*) menggunakan JSON Web Token (JWT), tanpa melibatkan integrasi sistem autentikasi pihak ketiga seperti *Single Sign-On (SSO)*.
- e. Pengujian keamanan sistem difokuskan pada evaluasi mekanisme otorisasi dan isolasi data antar tenant (*cross tenant data leakage*) pada tingkat perangkat lunak, tanpa melibatkan pengujian keamanan pada lapisan jaringan (*network penetration testing*).

D. Tujuan Penelitian

Tujuan Penelitian ini adalah :

- a. Mengetahui hasil perancangan dan implementasi sistem manajemen tugas berbasis *multi-tenant* sebagai solusi terhadap proses pengelolaan tugas konvensional di PT Bharata Internasional Pharmaceutical.
- b. Mengetahui hasil pengujian mekanisme otorisasi *Dynamic Context-Aware Role-Based Access Control* (RBAC) berbasis tiga variabel konteks, yaitu *ownership*, *resource state*, dan *tenant boundary*, dalam mengelola otorisasi dan isolasi data antar divisi pada sistem manajemen tugas berbasis *multi-tenant* di PT Bharata Internasional Pharmaceutical.

E. Manfaat Penelitian

Adapun manfaat yang diharapkan dari hasil penelitian ini dapat diklasifikasikan ke dalam dua aspek penerima manfaat utama, yaitu :

- a. Manfaat bagi perusahaan (Praktis)

Penelitian ini memberikan kontribusi praktis bagi PT. Bharata Internasional Pharmaceutical dalam bentuk rancangan dan implementasi sistem backend manajemen tugas *multi-tenant* berbasis *web*. Sistem ini diharapkan mampu membantu perusahaan dalam menjaga isolasi data antar divisi, mencegah akses tidak sah serta meningkatkan efektivitas pengelolaan tugas melalui sistem yang terintegrasi dan terstruktur.

- b. Manfaat bagi akademisi dan pengembang perangkat lunak (praktis dan teoritis).

Hasil penelitian ini dapat menjadi referensi bagi akademisi dan pengembang perangkat lunak dalam memahami penerapan arsitektur backend berbasis *Clean Architecture*. Selain itu, penelitian ini juga memberikan gambaran implementasi sistem *multi-tenant* dengan pendekatan *shared schema* pada basis data NoSQL (MongoDB), serta penerapan mekanisme keamanan berbasis *Dynamic Context-Aware RBAC* dalam pengelolaan hak akses sistem.