

# **OPTIMASI METODE YATA CRDT PADA REAL-TIME COLLABORATIVE TEXT EDITOR BERBASIS CLIENT- SERVER**

**Diajukan Sebagai Syarat Untuk Menyelesaikan  
Pendidikan Program Strata-1  
Pada Jurusan Teknik Informatika**



Oleh :

**Kemas Muhammad Husein Alviansyah  
NIM : 09021381924093**

**Jurusan Teknik Informatika  
FAKULTAS ILMU KOMPUTER UNIVERSITAS SRIWIJAYA  
2023**

## LEMBAR PENGESAHAN SKRIPSI

### OPTIMASI METODE YATA CRDT PADA REAL-TIME COLLABORATIVE TEXT EDITOR BERBASIS CLIENT- SERVER

Oleh :

**Kemas Muhammad Husein Alviansyah**  
NIM : 09021381924093

Palembang, 28 Juli 2023

Pembimbing I



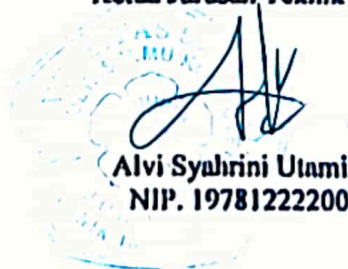
**Mastura Diana Marieska., M.T.**  
NIP. 198603212018032001

Pembimbing II



**Muhammad Qurhanul Rizqie., M.T., Ph.D.**  
NIP. 198712032022031006

Mengetahui,  
Ketua Jurusan Teknik Informatika



**Alvi Syahrini Utami., M.Kom.**  
NIP. 197812222006042003

## TANDA LULUS UJIAN KOMPREHENSIF

Pada hari kamis tanggal 27 juli 2023 telah dilaksanakan ujian komprehensif skripsi oleh Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya

Nama : Kemas Muhammad Husein Alviansyah

NIM : 09021381924093

Judul : Optimasi Metode YATA CRDT Pada Real-Time Collaborative Text Editor Berbasis Client-Server

dan dinyatakan LULUS.

1. Ketua Penguji

Samsuryadi., M.Kom., Ph.D.

NIP. 197102041997021003



2. Penguji I

Julian Supardi., M.T., Ph.D.

NIP. 197207102010121001



3. Pembimbing I

Mastura Diana Marieska., M.T.

NIP. 198603212018032001



4. Pembimbing II

Muhammad Ourhanul Rizqie., M.T., Ph.D.

NIP. 198712032022031006



Mengetahui  
Ketua Jurusan Teknik Informatika



Alvi Syahrini Utami., M.Kom.

NIP. 197812222006042003

## HALAMAN PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Kemas Muhammad Husein Alviansyah

NIM : 09021381924093

Program Studi : Teknik Informatika

Judul Skripsi : Optimasi Metode YATA CRDT Pada Real-Time Collaborative  
Text Editor Berbasis Client-Server

Hasil Pengecekan Software iThenticate/Turnitin: 1%

Menyatakan bahwa benar hasil laporan skripsi saya merupakan hasil karya sendiri dan bukan hasil penjiplakan / plagiat. Apabila ditemukan unsur penjiplakan / plagiat dalam laporan skripsi ini, maka saya bersedia menerima sanksi akademik dari Universitas Sriwijaya sesuai dengan ketentuan yang berlaku.

Demikian, pernyataan ini saya buat dengan sebenarnya dan tidak ada paksaan oleh siapapun.



Palembang, 29 Juli 2023

Kemas Muhammad Husein Alviansyah  
NIM. 09021381924093

## **MOTTO DAN PERSEMBAHAN**

“to the stars through difficulties”

Skripsi ini kupersembahkan kepada :

- Allah Subhanahu wa Ta'ala
- Orang Tua, Kakak, serta Keluargaku
- Dosen dan Guru – Guruku
- Teman – Temanku
- Fakultas Ilmu Komputer Universitas  
Sriwijaya

## **ABSTRACT**

Collaborative text editor systems are widely used to allow users to edit the same document simultaneously. YATA CRDT (Yet Another Transformation Approach Conflict-free Replicated Data Types) is an approach that can be implemented in these systems to ensure consistency across a large number of replicas. In YATA CRDT, the garbage collection mechanism can be a solution to the problem of the use of tombstone approach, which contributes to loss of performance as insertion operations increase. However, the garbage collection mechanism from previous research is not very good at guaranteeing consistency. In this research we developed a client-server based garbage collection and recovery mechanism that can guarantee consistency, a client-server based real-time collaborative text editor application system with the application of this mechanism, and conducting benchmark tests on the speed of execution time of YATA CRDT operations with different garbage collection mechanism configurations and observing the number of operations. The results show that the YATA CRDT method with the application of the client-server based garbage collection mechanism obtains a better operation execution time performance, and this mechanism is much more effective. There are still many things that can be done to increase the potential of the mechanism.

**Keywords :** YATA CRDT, Garbage Collection, Consistency, Client-server

## ABSTRAK

Sistem *collaborative text editor* secara luas digunakan untuk memungkinkan pengguna mengedit dokumen yang sama secara bersamaan. *YATA CRDT (Yet Another Transformation Approach Conflict-free Replicated Data Types)* merupakan pendekatan yang dapat diimplementasikan pada sistem ini untuk memastikan konsistensi pada seluruh replika dalam jumlah yang besar. Pada *YATA CRDT* mekanisme *garbage collection* mampu menjadi solusi pada masalah tidak dapat dibuangnya operasi *tombstone* yang berkontribusi pada penurunan kinerja seiring bertambahnya operasi *insertion*. Namun, mekanisme *garbage collection* dari penelitian sebelumnya belum baik dalam menjamin konsistensi. Pada penelitian ini dikembangkan mekanisme *garbage collection* dan *recovery* berbasis *client-server* yang dapat menjamin konsistensi, mengembangkan sistem aplikasi *real-time collaborative text editor* berbasis *client-server* dengan penerapan mekanisme ini, dan melakukan pengujian *benchmark* pada kecepatan waktu eksekusi operasi *YATA CRDT* dengan konfigurasi mekanisme *garbage collection* yang berbeda terhadap kasus pengujian yang ditentukan serta melakukan observasi pada jumlah operasi. Didapatkan hasil yang menunjukkan metode *YATA CRDT* dengan penerapan mekanisme *garbage collection* berbasis *client-server* memperoleh performa kecepatan waktu eksekusi operasi yang lebih unggul, dan mekanisme ini dapat bekerja walaupun dokumen sedang di modifikasi. Masih banyak hal yang dapat dilakukan untuk meningkatkan potensi dari mekanisme yang telah dikembangkan.

**Kata Kunci :** *YATA CRDT, Garbage Collection, Konsistensi, Client-server*

## KATA PENGANTAR

Puji dan syukur dipanjatkan kepada Allah SWT atas berkat, rahmat, karunia dan hidayah-Nya sehingga tugas akhir yang berjudul “*Optimasi Metode YATA CRDT Pada Real-Time Collaborative Text Editor Berbasis Client-Server*” dapat penulis selesaikan dengan baik. Karya tulis ini disusun untuk memenuhi salah satu persyaratan kelulusan studi tingkat Strata-1 pada Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya. Penulis juga menyampaikan banyak ungkapan terima kasih yang tak terhingga kepada:

1. Allah SWT atas berkah, rahmat, hidayah, dan karunia-Nya.
2. Mama tersayang Ida Siti Zuraidah., S.E., dan Papa tercinta Alm. Kemas Abdullah Husin atas do’a, dukungan, dan kesabarannya.
3. Kakak tersayang, Nyimas Irhamna D. F., S.Psi dan Ulil Amri., S.T. atas do’a dan dukungannya.
4. Ibu Alvi Syahrini Utami., M.Kom. atas bantuan dan jasanya selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya.
5. Ibu Mastura Diana Marieska., M.T. atas inspirasi, arahan, bimbingan, bantuan, serta dukungan selaku Dosen Pembimbing 1.
6. Bapak Muhammad Qurhanul Rizqie., M.T., Ph.D. atas arahan, bimbingan, bantuan, serta dukungan selaku Dosen Pembimbing 2.
7. Bapak DR. Ali Ibrahim, M. T. selaku Pembimbing Akademik selama proses perkuliahan, yang selalu membimbing, dan memberikan masukan kepada penulis.



8. Seluruh Dosen Jurusan Teknik Informatika serta Fakultas Ilmu Komputer Universitas Sriwijaya atas jasa, dan bantuannya.
9. Staf dan Pegawai Jurusan Teknik Informatika serta Fakultas Ilmu Komputer Universitas Sriwijaya atas jasa, dan bantuannya.
10. Teman-teman dan sahabat sirkel FOVV, Raihan, Fadel, Bintang, Asyraf, Nilam, Zafira, Shabrina, Rani, Reyhani, Aulia, Tarisa, dan Nurul atas inspirasi, dukungan, bantuan, dan perjuangan bersama.
11. Rekan-rekan *Google Development Student Club* Universitas Sriwijaya, dan Tim *DevOps* PT Tokopedia beserta rekan magang atas inspirasi, motivasi, dukungan, serta bantuannya.
12. Dan untuk semua teman-teman dan pihak lainnya yang tidak dapat disebutkan satu-persatu.

Terakhir, Penulis sangat menyadari bahwa masih terdapat banyak kekurangan pada proses maupun hasil pada penulisan karya tulis tugas akhir ini. Untuk itu kritik dan saran yang konstruktif sangat penulis apresiasi dan harapkan dalam penyempurnaan tugas akhir ini. Semoga karya tulis ini dapat bermanfaat dan berdampak kebaikan bagi banyak pihak dan penulis.

Palembang, 20 Juli 2023



Kemas Muhammad Husein Alviansyah

## DAFTAR ISI

<b>LEMBAR PENGESAHAN SKRIPSI .....</b>	<b>ii</b>
<b>TANDA LULUS UJIAN KOMPREHENSIF .....</b>	<b>iii</b>
<b>HALAMAN PERNYATAAN.....</b>	<b>iv</b>
<b>MOTTO DAN PERSEMBAHAN.....</b>	<b>v</b>
<b>ABSTRACT .....</b>	<b>vi</b>
<b>ABSTRAK .....</b>	<b>vii</b>
<b>KATA PENGANTAR.....</b>	<b>viii</b>
<b>DAFTAR ISI.....</b>	<b>x</b>
<b>DAFTAR GAMBAR.....</b>	<b>xiii</b>
<b>DAFTAR ALGORITMA .....</b>	<b>xv</b>
<b>DAFTAR TABEL .....</b>	<b>xvi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>I-1</b>
1.1.    Pendahuluan .....	I-1
1.2.    Latar Belakang .....	I-1
1.3.    Rumusan Masalah .....	I-3
1.4.    Tujuan Penelitian.....	I-4
1.5.    Manfaat Penelitian.....	I-4
1.6.    Batasan Masalah.....	I-5
1.7.    Sistematika Penulisan.....	I-5
1.8.    Kesimpulan.....	I-6
<b>BAB II KAJIAN LITERATUR .....</b>	<b>II-1</b>
2.1.    Pendahuluan .....	II-1
2.2.    Landasan Teori .....	II-1
2.2.1.    Aplikasi Kolaboratif.....	II-1
2.2.2.    CRDT .....	II-2
2.2.3.    YATA CRDT .....	II-4
2.2.3.1.    Unique Identifiers .....	II-4
2.2.3.2.    Operations .....	II-5
2.2.3.3.    Insert Algorithm .....	II-6
2.2.3.4.    Garbage Collection .....	II-9
2.2.3.5.    Offline Editing Support.....	II-10
2.2.4.    Model Client-Server .....	II-10

2.2.5.	<i>Sockets</i> .....	II-13
2.2.6.	<i>Threads</i> .....	II-15
2.2.7.	<i>Rational Unified Process</i> .....	II-16
2.2.8.	<i>Microbenchmarking</i> .....	II-19
2.3.	Penelitian Lain Yang Relevan .....	II-20
2.3.1.	<i>Memory Efficient CRDTs in Dynamic Environments</i> .....	II-20
2.3.2.	<i>Garbage Collected CRDTs on the Web</i> .....	II-20
2.4.	Kesimpulan .....	II-22
<b>BAB III METODOLOGI PENELITIAN .....</b>		<b>III-1</b>
3.1.	Pendahuluan .....	III-1
3.2.	Pengumpulan Data .....	III-1
3.2.1.	Jenis Data .....	III-1
3.2.2.	Sumber Data .....	III-2
3.2.3.	Metode Pengumpulan Data .....	III-2
3.3.	Tahapan Penelitian .....	III-2
3.3.1.	Perancangan dan Analisis Model Sistem .....	III-3
3.3.2.	Pembangunan Sistem .....	III-16
3.3.3.	Pengujian Sistem .....	III-16
3.3.4.	Analisis Hasil Pengujian dan Penarikan Kesimpulan .....	III-20
3.3.5.	Menuliskan Hasil Penelitian .....	III-21
3.4.	Metode Pengembangan Perangkat Lunak .....	III-21
3.4.1.	<i>Fase Inception</i> .....	III-21
3.4.2.	<i>Fase Elaboration</i> .....	III-21
3.4.3.	<i>Fase Construction</i> .....	III-22
3.4.4.	<i>Fase Transition</i> .....	III-22
3.5.	Manajemen Proyek Penelitian .....	III-22
3.6.	Kesimpulan .....	III-24
<b>BAB IV PENGEMBANGAN PERANGKAT LUNAK .....</b>		<b>IV-1</b>
4.1.	Pendahuluan .....	IV-1
4.2.	<i>Rational Unified Process</i> .....	IV-1
4.2.1.	<i>Fase Inception</i> .....	IV-1
4.2.2.	<i>Fase Elaboration</i> .....	IV-2
4.2.3.	<i>Fase Construction</i> .....	IV-22

4.2.4.	Fase <i>Transition</i> .....	IV-26
4.3.	Kesimpulan.....	IV-27
<b>BAB V</b>	<b>HASIL DAN ANALISIS PENELITIAN .....</b>	<b>V-1</b>
5.1.	Pendahuluan .....	V-1
5.2.	Data Hasil Percobaan/Penelitian .....	V-1
5.2.1.	Konfigurasi Percobaan .....	V-1
5.2.1.1.	Skenario Pengujian .....	V-1
5.2.1.2.	Data Pengujian.....	V-2
5.2.2.	Hasil Dan Analisis Per Kasus Pengujian .....	V-5
5.2.2.1.	Kasus Pengujian Riwayat Edit.....	V-5
5.2.2.2.	Kasus Pengujian Eksekusi Operasi Di Belakang Kata .....	V-9
5.2.2.3.	Kasus Pengujian Eksekusi Operasi Di Depan Kata .....	V-13
5.2.2.4.	Kasus Pengujian Eksekusi Operasi <i>Random</i> .....	V-17
5.3.	Analisis Pengujian .....	V-20
5.4.	Kesimpulan.....	V-22
<b>BAB VI</b>	<b>KESIMPULAN DAN SARAN.....</b>	<b>VI-1</b>
6.1.	Pendahuluan .....	VI-1
6.2.	Kesimpulan.....	VI-1
6.3.	Saran.....	VI-2
<b>DAFTAR PUSTAKA.....</b>		<b>xvii</b>

## DAFTAR GAMBAR

Gambar II-1. Contoh integrasi pada tipe teks (Nicolaescu et al., 2016). .....	II-6
Gambar II-2. Interaksi umum antara <i>client</i> C dan <i>server</i> S (M. van Steen & A.S. Tanenbaum, 2023). .....	II-11
Gambar II-3. Ilustrasi sistem <i>publish</i> dan <i>subscribe</i> pada <i>PS-CRDT</i> (Barreto et al., 2023). .....	II-12
Gambar II-4. Komunikasi <i>connection-oriented</i> menggunakan <i>socket</i> (M. van Steen & A.S. Tanenbaum, 2023). .....	II-13
Gambar II-5. Model <i>dispatcher / worker</i> untuk <i>multithreaded server</i> (M. van Steen & A.S. Tanenbaum, 2023). .....	II-15
Gambar II-6. Arsitektur metode <i>RUP</i> . .....	II-17
Gambar II-7. Ilustrasi algoritma penentuan stabilitas kausal (Rehn, 2020). .....	II-21
Gambar III-1. Tahapan Penelitian. ....	III-3
Gambar III-2. Diagram high level sistem keseluruhan. ....	III-4
Gambar III-3. Ilustrasi operasi yang tidak dapat dibuang. ....	III-5
Gambar III-4. Visualisasi tingkatan operasi. ....	III-6
Gambar III-5. Ilustrasi spesifikasi dan pengelompokkan operasi yang dibuang. ....	III-7
Gambar III-6. Ilustrasi algoritma <i>garbage collection client-server</i> . ....	III-14
Gambar IV-1. Diagram Arsitektur Sistem. ....	IV-3
Gambar IV-2. Diagram <i>Use Case</i> . ....	IV-4
Gambar IV-3. <i>Activity Diagram</i> Membagikan Dokumen. ....	IV-13
Gambar IV-4. <i>Activity Diagram</i> Mengakses Dokumen. ....	IV-14
Gambar IV-5. <i>Activity Diagram</i> Menyimpan Dokumen Lokal. ....	IV-15
Gambar IV-6. <i>Activity Diagram</i> Membuka Dokumen Lokal. ....	IV-15
Gambar IV-7. <i>Activity Diagram</i> Membuat Dokumen Baru. ....	IV-16
Gambar IV-8. <i>Sequence Diagram</i> Membagikan Dokumen. ....	IV-17
Gambar IV-9. <i>Sequence Diagram</i> Mengakses Dokumen. ....	IV-18
Gambar IV-10. <i>Sequence Diagram</i> Menyimpan Dokumen Lokal. ....	IV-19
Gambar IV-11. <i>Sequence Diagram</i> Membuka Dokumen Lokal. ....	IV-19
Gambar IV-12. <i>Sequence Diagram</i> Membuat Dokumen Baru. ....	IV-19
Gambar IV-13. <i>Class Diagram CRDT</i> dan Komponen <i>Network</i> . ....	IV-20
Gambar IV-14. <i>Class Diagram</i> Sistem <i>Client</i> . ....	IV-21
Gambar IV-15. <i>Class Diagram</i> Sistem <i>Server</i> . ....	IV-22
Gambar V-1. Grafik Pengujian Benchmark Waktu Eksekusi Pada Kasus Riwayat Pengeditan Dunia Nyata. ....	V-6
Gambar V-2. Grafik Observasi Jumlah Operasi Pada Kasus Riwayat Pengeditan Dunia Nyata. ....	V-7
Gambar V-3. Grafik Pengujian Benchmark Waktu Eksekusi Pada Kasus Eksekusi Operasi Di Belakang Kata. ....	V-10
Gambar V-4. Grafik Observasi Jumlah Operasi Pada Kasus Eksekusi Operasi Di Belakang Kata. ....	V-11
Gambar V-5. Grafik Pengujian Benchmark Waktu Eksekusi Pada Kasus Eksekusi Operasi Di Depan Kata. ....	V-14

Gambar V-6. Grafik Observasi Jumlah Operasi Pada Kasus Eksekusi Operasi Di Depan Kata.....	V-14
Gambar V-7. Grafik Pengujian Benchmark Waktu Eksekusi Pada Kasus Eksekusi Operasi <i>Random</i> .....	V-18
Gambar V-8. Grafik Observasi Jumlah Operasi Pada Kasus Eksekusi Operasi <i>Random</i> .....	V-18

## **DAFTAR ALGORITMA**

Algoritma II-1: Insert Algorithm .....	II-8
Algoritma III-1: Operation Specification Functions .....	III-9
Algoritma III-2: Garbage Collection Algorithm .....	III-10
Algoritma III-3: Recovery Algorithm .....	III-11
Algoritma III-4: Find Affected GC Algorithm .....	III-12
Algoritma III-5: Find GC Items Algorithm .....	III-13

## DAFTAR TABEL

Tabel III-1. Tabel alat bantu pada penelitian. ....	III-15
Tabel III-2. Tabel pengukuran perbandingan jumlah operasi dan waktu eksekusi. .....	III-20
Tabel III-3. <i>Gantt Chart</i> rencana proses penelitian dan pengembangan perangkat lunak. ....	III-23
Tabel IV-1. Spesifikasi <i>Actor</i> . ....	IV-4
Tabel IV-2. Spesifikasi <i>Use Case</i> . ....	IV-5
Tabel IV-3. Skenario <i>Use Case</i> Membuat Dokumen Baru. ....	IV-6
Tabel IV-4. Skenario <i>Use Case</i> Membagikan Dokumen. ....	IV-6
Tabel IV-5. Skenario <i>Use Case</i> Mengakses Dokumen. ....	IV-8
Tabel IV-6. Skenario <i>Use Case</i> Menyimpan Dokumen Lokal. ....	IV-9
Tabel IV-7. Skenario <i>Use Case</i> Membuka Dokumen Lokal. ....	IV-10
Tabel IV-8. Skenario <i>Use Case</i> Menyimpan Dokumen Yang Dibagikan. ....	IV-11
Tabel IV-9. Skenario <i>Use Case</i> Menyebarkan Modifikasi Dokumen. ....	IV-12
Tabel IV-10. Tabel Hasil Analisa Kebutuhan Fungsionalitas dan Non-Fungsionalitas Pada Sistem. ....	IV-26
Tabel V-1. Sampel Data Kasus Pengujian Eksekusi Operasi Berdasarkan Dataset Riwayat Pengeditan Dunia Nyata. ....	V-3
Tabel V-2. Sampel Data Kasus Pengujian Eksekusi Operasi Di Belakang Kata. V-3	
Tabel V-3. Sampel Data Kasus Pengujian Eksekusi Operasi Di Depan Kata. ...	V-4
Tabel V-4. Sampel Data Kasus Pengujian Eksekusi Operasi <i>Random</i> Dengan Posisi <i>Random</i> . ....	V-5
Tabel V-5. Sampel Data Dan Perbandingan Hasil Pengujian Pada Kasus Riwayat Pengeditan Dunia Nyata. ....	V-5
Tabel V-6. Sampel Data Dan Perbandingan Hasil Pengujian Pada Kasus Eksekusi Operasi Di Belakang Kata. ....	V-9
Tabel V-7. Sampel Data Dan Perbandingan Hasil Pengujian Pada Kasus Eksekusi Operasi Di Depan Kata. ....	V-13
Tabel V-8. Sampel Data Dan Perbandingan Hasil Pengujian Pada Kasus Eksekusi Operasi <i>Random</i> . ....	V-17



# BAB I

## PENDAHULUAN

### 1.1. Pendahuluan

Dalam bab pendahuluan ini diuraikan tentang pokok-pokok pikiran yang melandasi rencana skripsi. Pokok-pokok pikiran dimaksud antara lain latar belakang masalah penelitian, perumusan masalah/permasalahan penelitian, tujuan penelitian dan manfaat penelitian.

### 1.2. Latar Belakang

Sistem *collaborative text editing* sering mereplikasi dokumen pada *site* yang terdistribusi secara geografis atau di perangkat pengguna untuk menjamin ketersediaan yang tinggi dan responsif (Attiya et al., 2021). fitur penting dari sistem pengeditan dokumen kolaboratif adalah bahwa semua perubahan pada akhirnya disebarkan ke semua replika dan dimasukkan ke dalam dokumen secara konsisten. Secara khusus, sistem seperti itu bertujuan untuk menjamin *eventual consistency*.

Hal ini memungkinkan *state* pada replika dapat berbeda untuk sementara dan diperlukan suatu mekanisme untuk menggabungkan modifikasi konkuren menjadi satu *state* yang sama. *CRDT (Conflict-free Replicated Data Types)* merupakan pendekatan yang bertujuan mengatasi masalah ini, *CRDT* adalah tipe data abstrak yang dirancang untuk dapat direplikasi pada beberapa proses (Preguiça et al., 2018). *CRDT* telah dikembangkan untuk berbagai tipe data yang berbeda, salah satunya adalah tipe data *list* atau *sequence*, *list* yang direplikasi dapat digunakan untuk aplikasi *collaborative text editor*, karena pada dasarnya teks adalah kumpulan atau *list* dari karakter (Kleppmann, 2020).

Salah satu metode *list CRDT* adalah, *YATA (Yet Another Transformation Approach) CRDT* (Nicolaescu et al., 2016). *YATA CRDT* merepresentasikan data linear sebagai sebuah *doubly linked list*, dengan dua jenis operasi yaitu *insert* atau *insertion* yang merupakan representasi dari elemen pada *list*, dan operasi *delete* untuk menghapus *insertion* atau elemen. *YATA* memiliki kelebihan dari ukuran pesan yang kecil, mendukung pengeditan *offline*, dan tipe data yang *extendable*, namun perlu diketahui *YATA* menggunakan *tombstone* pada operasi *delete*, yang berarti elemen tidak dibuang dari *list* melainkan hanya ditandai sebagai dihapus. Dengan mempertahankan elemen yang sudah dihapus menyebabkan ukuran dari dokumen pada setiap replika akan terus meningkat dan berakibat performa yang terus menurun.

*YATA* mengatasi masalah penurunan performa dengan menambahkan mekanisme *garbage collector* berbasis waktu pada setiap replika, untuk menghindari peningkatan operasi yang tidak diperlukan lagi. Namun, menambahkan mekanisme ini dalam beberapa kasus dapat merusak mekanisme penggabungan yang konkuren, karena memungkinkan sebuah replika dapat membuat elemen yang memiliki referensi ke elemen yang sudah dibuang pada replika lain, hal ini menyebabkan konsistensi tidak terjamin.

Pada penelitian ini akan dikembangkan mekanisme *garbage collection* berbasis *client-server* untuk membantu mengurangi *tombstone* dan dapat menjamin konsistensi. *Client-server* adalah sebuah model arsitektur sistem yang terdiri dari dua bagian, sistem *client* dan sistem *server*, keduanya berkomunikasi melalui jaringan komputer (Kumar, 2019).

Pada mekanisme yang dikembangkan ini, pembagian sistem *client-server* adalah, satu replika sebagai *server* yang tersentralisasi, dan replika lain yang digunakan oleh pengguna sebagai *client*, *server* sentral berperan sebagai pengelola *garbage collector* dan *recovery*. Sedangkan *client* yang merupakan antarmuka pengguna akan menerima dan melakukan aksi yang diberikan oleh *server*. Kedua *client* dan *server* sentral juga tetap berperan sebagai replika *YATA CRDT*. Replika *YATA CRDT* disesuaikan agar dapat mendukung operasi *garbage collection* dan *recovery*, penyesuaian dilakukan tanpa menghilangkan atau mengubah basis dari komponen replika *YATA CRDT*.

Penggunaan komponen tersentralisasi pada *CRDT* memberikan kelebihan untuk memfasilitasi fitur persistensi (Barreto et al., 2023). Dengan ini, *server* sentral dapat dimanfaatkan untuk menyimpan semua operasi, dan operasi ini dapat digunakan pada mekanisme *recovery* untuk menjamin konsistensi pada replika *YATA CRDT* di sisi *client*, agar *client* dapat tetap menerima operasi yang memiliki referensi ke operasi yang sudah dibuang oleh *garbage collector*. Terutama, operasi yang berasal dari replika yang melakukan modifikasi saat *offline*.

Mekanisme *garbage collection* ini diharapkan dapat mengatasi kekurangan dari implementasi *garbage collector YATA CRDT* yang sebelumnya dan menghasilkan performa yang lebih optimal pada *YATA CRDT*, terutama pada sisi *client* sebagai antarmuka pengguna dari aplikasi *collaborative text editor*.

### 1.3. Rumusan Masalah

Rumusan masalah utama dalam penelitian ini adalah:

Bagaimana cara meningkatkan performa *YATA CRDT* dengan menerapkan mekanisme *garbage collection* berbasis *client-server* yang dikembangkan pada penelitian ini?

Untuk menjawab rumusan masalah utama tersebut, terdapat pertanyaan pendukung penelitian ini, antara lain :

1. Bagaimana implementasi dari mekanisme *garbage collection YATA CRDT* berbasis *client-server*?
2. Seberapa tinggi peningkatan performa *YATA CRDT* menggunakan mekanisme *garbage collection* berbasis *client-server* dalam hal waktu eksekusi?

#### **1.4. Tujuan Penelitian**

Terdapat tujuan dari penelitian ini, antara lain :

1. Mengembangkan sebuah mekanisme *garbage collection YATA CRDT* berbasis *client-server* untuk meningkatkan performa dan mengatasi kekurangan dari mekanisme *garbage collection* pada penelitian sebelumnya dalam menjamin konsistensi.
2. Mengukur seberapa tinggi peningkatan performa mekanisme *garbage collection YATA CRDT* berbasis *client-server* dalam hal waktu eksekusi.

#### **1.5. Manfaat Penelitian**

Terdapat manfaat dari penelitian ini, antara lain :

1. Meningkatkan performa *YATA CRDT* menggunakan mekanisme *garbage collection* dengan model *client-server*.

2. Hasil penelitian dapat digunakan sebagai referensi untuk penelitian pada topik *CRDT* ataupun menjadi bahan referensi untuk literatur lainnya.

#### **1.6. Batasan Masalah**

Batasan masalah dalam penelitian ini, antara lain :

1. *Text editor* pada aplikasi hanya memodifikasi karakter teks pada dokumen, tidak termasuk mengubah font atau gaya teks seperti *rich text editor*.
2. Aplikasi dikembangkan menggunakan protokol dasar *TCP/IP* dengan *Socket*.
3. Melakukan pengukuran performa pada waktu eksekusi operasi *YATA CRDT* pada sisi *client*, tidak termasuk waktu latensi pada jaringan.
4. Pengujian pada sistem *client* dan *server* disimulasikan pada satu perangkat menggunakan jaringan lokal.

#### **1.7. Sistematika Penulisan**

Sistem penulisan tugas akhir ini sesuai dengan standar penulisan tugas akhir Fakultas Ilmu Komputer Universitas Sriwijaya yaitu :

### **BAB I PENDAHULUAN**

Bab ini berisi latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, sistematika penulisan, dan kesimpulan dari isi bab.

### **BAB II KAJIAN LITERATUR**

Bab ini berisi kajian literatur yang menjelaskan landasan teori, penelitian lain yang relevan, dan kesimpulan dari isi bab. Hal yang dijelaskan pada bab ini antara lain mengenai aplikasi kolaboratif atau

*Groupware Software*, metode *YATA CRDT*, model *Client-Server*, *Threads*, *Sockets*, metode pengembangan perangkat lunak *RUP (Rational Unified Process)*, *microbenchmarking*, dan penelitian yang relevan seperti *CRDT Eager Garbage Collection*.

### **BAB III METODOLOGI PENELITIAN**

Bab ini berisi langkah – langkah dan rencana dari penelitian dan pengujian yang dijelaskan secara rinci. Bab ini juga mencakup perancangan dan analisis model sistem, serta pengusulan rancangan mekanisme *garbage collection* berbasis *client-server*.

### **BAB IV PENGEMBANGAN PERANGKAT LUNAK**

Bab ini menguraikan proses pengembangan perangkat lunak, digunakan metode *RUP*, setiap proses diuraikan berdasarkan fasenya antara lain, fase *inception*, fase *elaboration*, fase *construction*, dan fase *transition*.

### **BAB V HASIL DAN ANALISIS PENELITIAN**

Bab ini menjabarkan hasil evaluasi pengujian penelitian, dan analisa hasil untuk mendapatkan kesimpulan dari penelitian ini.

### **BAB VI KESIMPULAN DAN SARAN**

Bab ini terdapat kesimpulan yang di tarik dari seluruh hasil proses penelitian, serta saran yang diharapkan berguna untuk penelitian lain kedepannya.

#### **1.8. Kesimpulan**

Pada Bab 1 ini dapat disimpulkan penelitian ini bertujuan untuk memberikan solusi pada masalah utama yaitu bagaimana cara meningkatkan

performa *YATA CRDT* tanpa menimbulkan masalah pada konsistensi, dengan mengembangkan dan menerapkan mekanisme *garbage collection YATA CRDT* berbasis *client-server* untuk meningkatkan performa dan mengatasi kekurangan dari mekanisme *garbage collection* penelitian sebelumnya.

## DAFTAR PUSTAKA

- Attiya, H., Burckhardt, S., Gotsman, A., Morrison, A., Yang, H., & Zawirski, M. (2021). Specification and space complexity of collaborative text editing. *Theoretical Computer Science*, 855, 141–160. <https://doi.org/10.1016/J.TCS.2020.11.046>
- Barreto, A., Paulino, H., Silva, J. A., & Preguiça, N. (2023). PS-CRDTs: CRDTs in highly volatile environments. *Future Generation Computer Systems*, 141, 755–767. <https://doi.org/10.1016/J.FUTURE.2022.12.013>
- Bauwens, J., & Boix, E. G. (2019). Memory efficient CRDTs in dynamic environments. *VMIL 2019 - Proceedings of the 11th ACM SIGPLAN International Workshop on Virtual Machines and Intermediate Languages, Co-Located with SPLASH 2019*, 48–57. <https://doi.org/10.1145/3358504.3361231>
- Costa, D., Bezemer, C.-P., Leitner, P., & Andrzejak, A. (2021). What’s Wrong with My Benchmark Results? Studying Bad Practices in JMH Benchmarks. *IEEE Transactions on Software Engineering*, 47(7), 1452–1467. <https://doi.org/10.1109/TSE.2019.2925345>
- Hassan, S. A., & Abdelbaki, N. (2023). A Survey of Concurrency Control Algorithms in Collaborative Applications. In A. E. Hassanien, V. Snášel, M. Tang, T.-W. Sung, & K.-C. Chang (Eds.), *Proceedings of the 8th International Conference on Advanced Intelligent Systems and Informatics 2022* (pp. 309–320). Springer International Publishing.
- Jahns, K. (2022). *GitHub - dmonad/crdt-benchmarks: A collection of CRDT benchmarks*. <https://github.com/dmonad/crdt-benchmarks>
- Kleppmann, M. (2020). Moving Elements in List CRDTs. *Proceedings of the 7th Workshop on Principles and Practice of Consistency for Distributed Data*. <https://doi.org/10.1145/3380787.3393677>
- Kleppmann, M., & Beresford, A. R. (2016). *A Conflict-Free Replicated JSON Datatype*. <https://doi.org/10.1109/TPDS.2017.2697382>
- Kumar, S. (2019). A REVIEW ON CLIENT-SERVER BASED APPLICATIONS AND RESEARCH OPPORTUNITY. *International Journal of Scientific Research*, 10, 33857–33862. <https://doi.org/10.24327/ijrsr.2019.1007.3768>
- Litt, G., Lim, S., Kleppmann, M., & Van Hardenberg, P. (2022). Peritext: A CRDT for Collaborative Rich Text Editing. *Proceedings of the ACM on Human-Computer Interaction*, 6(CSCW2). <https://doi.org/10.1145/3555644>
- M. van Steen, & A.S. Tanenbaum. (2023). *Distributed Systems* (4th ed.). <https://www.distributed-systems.net/index.php/books/ds4/>



- Nicolaescu, P., Jahns, K., Derntl, M., & Klamma, R. (2016). Near Real-Time Peer-to-Peer Shared Editing on Extensible Data Types. *Proceedings of the 2016 ACM International Conference on Supporting Group Work*, 39–49. <https://doi.org/10.1145/2957276.2957310>
- Preguiça, N., Baquero, C., & Shapiro, M. (2018). *Conflict-free Replicated Data Types (CRDTs)*. [https://doi.org/10.1007/978-3-319-63962-8\\_185-1](https://doi.org/10.1007/978-3-319-63962-8_185-1)
- Rehn, M. (2020). *Garbage Collected CRDTs on the Web : Studying the Memory Efficiency of CRDTs in a Web Context*. <http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-413299>
- Saeed, S., Jhanjhi, N. Z., Naqvi, M., & Humayun, M. (2019). Analysis of software development methodologies. *International Journal of Computing and Digital Systems*, 8(5), 445–460. <https://doi.org/10.12785/IJCDS/080502>
- Sommerville, I. (2015). *Software Engineering* (10th ed.). Pearson.